HURRICANE SURGE FLOODING DAMAGE ASSESSMENT AND WEB-BASED GAME DEVELOPMENT TO SUPPORT K12 EDUCATION FOR UNDERSTANDING CLIMATE CHANGE IMPACT ON HURRICANE SURGE FLOODING DAMAGE

A Dissertation

by

CHIH-HUNG HSU

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Francisco Olivera
Committee Members,	Anthony Cahill
	James Kaihatu
	Steven Quiring
Head of Department,	Robin Autenrieth

August 2014

Major Subject: Civil Engineering

ABSTRACT

Hurricane surge floods have caused devastating damage along coastal areas all over the world. Yet many recent studies have shown that global warming could increase the hurricane flooding damage by hurricane intensification and sea level rise. Hence, the ability to predict surge elevations and to use this information for damage estimation is fundamental for saving lives and protecting property. In this study, we developed a framework that allows one to acquire information of hurricane flooding damage (e.g. structural damage, population being affected, flooded area, etc.) for different hurricanes. This framework is based on Surge Response Functions (SRFs). SRFs are physical scaling laws derived from a suite of discrete ADvanced CIRCulation Model (ADCIRC) simulations and represent surge values as a function of hurricane parameters. The advantage of this SRF-based approach is that a large number of storms can be efficiently evaluated and considered in the analysis, without losing accuracy in the surge estimates. To extrapolate the surge water elevation inland, SRF zones were defined within which the water elevation was considered horizontal. Individual parcel flood damage was calculated based on the flood water depth and damage vs. water depth curves included in the Federal Emergency Management Agency's (FEMA) application HAZUS. Parcel data (property value, population) and business data (employee size, and sales volume) are collected and used to conduct risk analysis under different future climate scenarios. Expected changes for future climate scenarios (i.e., IPCC scenarios B1, A1B and A1FI for the 2030's and 2080's) were considered by accounting for projected sea surface temperature increases and sea level rise, which modify the probability distribution of

hurricane central pressure and change the baseline of damage calculation, respectively. Flood risk estimates and maps are developed for Corpus Christi in Texas, and Gulfport in Mississippi. For the case of Corpus Christi, it was found that, as the projected sea surface temperature increased, higher surge values are more likely to occur, as expected, resulting in higher expected damage. The risk map of Port Aransas in the Corpus Christi area, for example, shows that the risk is in the range of 1% to 4% of the property value for current climate conditions, and shifts to 1% to 8% for the 2030's and 1% to 14% for the 2080's. The concept of the parameterized hurricane damage analysis is also used to construct a web-based game "VisHurricane" which is intent to be used as an educational tool for K-12 students to arise their attention of current issues on climate change and potential future hurricane surge flooding risk.

DEDICATION

To my mother and sisters

ACKNOWLEDGEMENTS

I would like to thank my committee chair, Dr. Francisco Olivera, and my committee members, Dr. Cahill, Dr. Kaihatu, and Dr. Quiring, for their guidance and support throughout the course of this research.

Thanks also to my friends, classmates and the department faculty and staff for making my time at Texas A&M University a great experience.

I also want to extend my gratitude to the National Sea Grant of the U.S. Department of Commerce's National Oceanic and Atmospheric Administration, for funding this research (Grant No. NA10OAR4170099).

Special thanks to Dr. Irish in Virginia Tech University, her former students Ikpoto Udoh, Nick Taylor, and Dr. Kaihatu's student Patrick Mclaughlin, for their support and suggestions on this research. Without them this work would not have been possible.

NOMENCLATURE

ADCIRC	Advanced Circulation model
ANOVA	Analysis of Variance
DFIRM	Digital Flood Insurance Rate Maps
FEMA	Federal Emergency Management Agency
FIA	Federal Insurance Administration
GIS	Geographical Information Systems
HAZUS	Hazards US Multi Hazard
IPCC	International Panel on Climate Change
MSL	Mean Sea-level
NOAA	National Oceanic and Atmosphere Administration
NRC	National Research Council
RMSE	Root Mean Square Error
SLR	Sea-level Rise
SWAN	Simulating Waver Near Shore
US	United States
USACE	United States Army Corps of Engineering
USGS	United States Geological Survey
L_{km}	Losses in parcel k of block j caused by hurricane m,
d_{km}	Water depth in parcel k caused by hurricane m, and
V_k	Improvement value in parcel k.
R_k	Flood risk for parcel k located

p_m	Probability of occurrence of hurricane m in a given year
λ	Probability that hurricanes would make landfall on a specified coastline range
ζ	Surge response
c_p	Hurricane central pressure
R_p	Hurricane radius to the maximum wind
v_f	Hurricane forward speed
θ	Hurricane approach angle
x_o	Hurricane landfall location
$T(x)$	Return period for surge value x
ζ_o	Current day surge response (without effect of SLR)
ζ_s	Projected surge response with effect of SLR
η_s	Water elevation with effect of SLR
k, l	adjustment coefficients for surge value
$p_{\Delta SST}$	Future projected hurricane central pressure
p_o	Current day (2000s) hurricane central pressure
ΔSST	Sea surface temperature change
Δp	Central pressure deficit (peripheral pressure minus p_o)

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF TABLES

Page

1. INTRODUCTION

Hurricanes are a major natural hazard to many coastal areas of the world. Land falling hurricanes generates strong winds, heavy rainfall and surges, which often cause enormous property damages and even deaths. In the United States, for example, recent hurricane events such as Hurricane Katrina (2005), Hurricane Ike (2008), Hurricane Wilma (2005) and Hurricane Rita (2005) have caused up to a hundred billion dollars of loss (USGS 2005; Lott and Ross, 2006). As the population becomes greater in coastal region (Hinrichsen, 1999; Wilson and Fischetti, 2010) and the increase of flood risk due to sea level rise and climate change (Knutson and Tuleya, 2004; Church et al., 2008; Nicholls and Cazenave, 2010), the assessment of flood frequency and its impact on population and properties loss has become critical for hazard management.

This research focuses on the assessment of the damage caused by hurricanes surges. Damage assessment is a major concern for society because it helps the authorities develop policies to manage or reduce them. For example, the US government invests a considerable amount of resources in developing floodplain maps through the Federal Emergency Management Agency's (FEMA) Digital Flood Insurance Rate Maps (DFIRM) program. The DFIRMs are used to regulate the areas in which structures can be built in order to reduce the damage when flooding occurs. FEMA also has developed a hazard damage assessment tool, Hazard US (HAZUS), which can roughly estimate the damage caused by different hazards such as earthquakes, floods and hurricanes.

Overall, the information needed for hazard mitigation and management includes mapping the flooded area, estimating the damage, and assessing the flood risk. A common approach to obtain this information consists of using hydrodynamic models to represent the surge for selected hurricane events; however, the computation burden for running hydrodynamic models is often an issue when a large number of storms need to be studied. Recent studies, using surge response functions and probability methods with optimal sampling (Irish et al., 2008b; Resio et al., 2009; Song et al., 2012; Udoh, 2012; Irish and Resio, 2013), have shown that this approach can be used for estimating surge values for large numbers of cases given the hurricane parameters without having to run hydrodynamic models for each case, and still keep the accuracy of the surge estimation.

This study combines the concepts of surge response functions, joint probability of the hurricane parameters and damage calculation, to develop a framework that allows one to efficiently assess the flood risk caused by hurricane events, and develop flood and risk maps to assist decision-makers on hurricane hazard management. Aside from the analysis product that assist decision making, public awareness of hurricane hazard is also of concern as a part of mitigation plans. As personal computers and the Internet are widely used in recent decades, many stand-alone or online applications have been developed for the purpose of educating the public on the mechanics of hurricane and preparedness against hurricane hazards (Luo et al., 2008; Natioal Geographic, 2012; UCAR, 2012); however, few of them have directly link hurricane characteristics and future climate conditions to the damage risk. Based on the concept of parameterized hurricane damage analysis, a web-based game “VisHurricane!” has been designed to show the impact of climate change on hurricane intensity and, consequently, hurricane surge induced damage. “VisHurricane!” is targeted to K-12 students.

2. BACKGROUND

Some studies approach the hurricane damage assessment by collecting high water marks and relating them to economic losses or investigating structure performance after hurricane events. For example, van de Lindt et al. (2007) investigated the damage from Hurricane Katrina on wood-frame structures due to wind and intrusion of water. Robertson et al. (2007) surveyed and categorized structural damage from hurricane surge during Hurricane Katrina. Yazdani et al. (2010) inspected more than 1,000 sites to study building performance, particularly residential and commercial structures, during different storm events. Fritz et al. (2008) collected surge height data and surveyed the effect of storm surges on built and natural environments from Hurricane Katrina.

Hydrodynamic models can be used to represent hurricanes and associated surges, and to estimate the generated damage. Some of these models include, the Sea, Lake, and Overland Surge from Hurricane (SLOSH) developed by the National Oceanic and Atmospheric Administration (NOAA, 2014); the Advanced Circulation (ADCIRC) model (Luettich and Westerink, 2004); the Eulerian-Lagrangian circulation (ELCIRC) model (Zhang et al., 2004; Baptista et al., 2005); the fully non-linear Finite Volume Coastal Ocean Model (FVCOM) (Chen et al., 2003); and the fine grid coastal surge model Curvilinear-grid Hydrodynamics in 3D (CH3D) (USACE, 2014). These models have been used in a number of studies, including Mattocks and Forbes (2008), Westerink et al. (2008), Rego and Li (2010), Westerink et al. (2008), Ebersole et al. (2010), and Xu et al. (2010); some other studies have also accounted for wind waves to storm surge modeling, including Chen et al. (2007), Huang et al. (2010), Dietrich et al. (2010), Bunya et al. (2010), Sheng et al. (2010), Church et al. (2004) , and Dietrich et al. (2011).

Hurricane-induced surge and wind damage can be assessed with two different approaches: a model-based approach and a survey-based approach. In the model-based approach, damage is estimated for either a hypothetical future event or for a past event. In the survey-based approach, direct evidence of damage is gathered and quantified. Damage assessed with the survey-based approach, additionally, can be used to calibrate and validate the model-based approach. However, with the model-based approach, to quantify the loss due to a hurricane given the corresponding surge and wind fields, further work is still needed. Li et al. (2012) developed a loss assessment methodology for combined wind and surge on residential buildings. In their study, the correlation between hurricane- and wind-induced surge is investigated using regressive analysis of historical data, the losses are estimated as a percentage of the value of the structures and items inside with respect to the surge elevation. Bjarnadottir et al. (2013) proposed a framework for hurricane loss assessment of residential buildings, which includes probabilistic models of hurricane wind speed and hurricane-induced surge and vulnerability functions to estimate damage cost. They also account for climate change effects on hurricane wind speed and frequency. Frey et al. (2010) use ADCIRC and SWAN to simulate hurricane surges, and then apply the damage function approach to estimate flood damage and population affected for the case of Corpus Christi, in Texas. Their study accounts for the impact of climate change by coupling the predicted sea surface temperature (SST) and sea level rise (SLR), from the Model for the Assessment of Greenhouse-gas Induced Climate Change / A regional climate SCENario GENerator (MAGICC⁄SCENGEN) (Wigley, 2008), to the hurricane intensity. Ferreira (2012) also use ADCIRC and SWAN to simulate surge for selected hurricane tracks. Upon considering the surge attenuation by costal wetland and impact of climate change on sea level rise and hurricane intensity, the damage, population and businesses affected by hurricane surge at census block level is evaluated. FEMA has also developed Hazards U.S. Multi-Hazard (HAZUS-MH) (Schneider and Schauer, 2006), a multi-hazard loss estimation software application that includes modules to estimate losses caused by earthquakes, wind and floods. The flood loss estimation module uses damage functions to estimate the flood damage of structures with respect to the flood depth.

Some of the mentioned studies, such as (Frey et al., 2010; Ferreira, 2012), use selected historical or hypothetical hurricanes to investigate hurricane damage; however, to assess the impact of a large number of hurricanes with different parameters (i.e. intensity, radius to maximum wind, landfall location), a large number of model simulations are needed. Resio et al. (2009) point out that numerical storm surge simulations are highly computationally intensive, especially when a large number of hurricane scenarios are needed to be investigated. To resolve this issue, they propose a joint probability method with optimal sampling (JPM-OS), and introduced a surge response function (SRF) approach as a means for optimally sampling the storm parameter population in order to reduce the computational requirement for surge modeling. The SRF approach and in combination with the JPM-OS have been implemented in other studies (Irish et al., 2011a; Mousavi et al., 2011; Irish et al., 2012; Song et al., 2012; Irish and Resio, 2013).

Owing to the Texas Sea Grant funded project “A Parameterized Climate Change Projection Model for hurricane Flooding, Wave Action, Economic Damages, and Population Dynamics”, Dr. Jennifer Irish group in the Department of Coastal Hazards and Engineering at Virginia Tech University, has developed SRFs at a number of locations along the coastline of Corpus Christi, TX, Gulfport, MS and Panama, FL. In this study, these SRFs are used to develop a hurricane surge flood damage assessment framework for these three cities. This framework will allow us to perform damage analysis at a parcel level in three cities, to delineate the flooded areas, and to estimate affected population/businesses for given hurricane parameters, in an efficient manner comparing to the event-based numerical modeling approach.

In this study, as well as in others mentioned above, damage assessment is conducted by estimating the flood depth at a given building and associating a percent structural damage to it. This percent damage increases as the flood depth increases and depends on the building type. The building damage is estimated as the product of the percent damage times the building improvement value (i.e., fraction of the property value corresponding to the structure.) The plots of percent damage vs. flood depth result in curves known as damage curves. In general, damage curves are derived based on post-event surveys, analyses of insurance claims, and historical flood data analyses (Nadal et al., 2010). A number of damage curves have been developed by the Federal Insurance Administration (FIA) and the U.S. Army Corps of Engineers (USACE) (Scawthorn et al., 2006), and are included in HAZUS-MH. Damage curves are commonly used in damage analysis because the concept is simple and can be easily applied. In this study, the damage curves contained in HAZUS-MH are used for damage value calculation.

SRFs are non-dimensional physics-based scaled equations derived from a suite of discrete ADCIRC simulations (Irish et al., 2008b; Song et al., 2012) and represent surge values as a function of hurricane parameters (i.e., central pressure, radius, forward speed, approaching angle and landfall location). These equations represent a continuous surface of surge response, which is used to predict the surge values for given hurricane parameters without running numerical model for specific cases. According to (Irish and Resio, 2013), the surge response model can be expressed as follows:

$$z_{max}(x) = \phi\left(x, p_0, R_p, v_f, \theta, x_o, MSL\right) + \varepsilon_z \tag{2.1}$$

$$\varepsilon_z^2 = \varepsilon_{tide}^2 + \varepsilon_{surge\ simulation}^2 + \varepsilon_{wave}^2 + \varepsilon_{wind}^2 + \cdots \tag{2.2}$$

where

z_{max} *is the surge response*

ϕ *is the surge response function*

x *is location of interest*

x_0 *is landfall location*

R_p *is hurricane radius to the maximum wind*

θ *is the hurricane approach angle with respect to the shoreline*

v_f *is the hurricane forward speed near landfall*

MSL *is the mean sea level*

ε_z *is the epistemic uncertainty in the surge response* (Resio et al., 2009; Resio et al., 2013)

Figure 2-1 shows an example of the locations that surge response functions are obtained. These locations are termed “SRF station” in this study, as SRF is short for “Surge Response Function”. Table 2–1 shows an example of how surge value is stored for different hurricane scenarios (i.e. different combination of hurricane parameters).

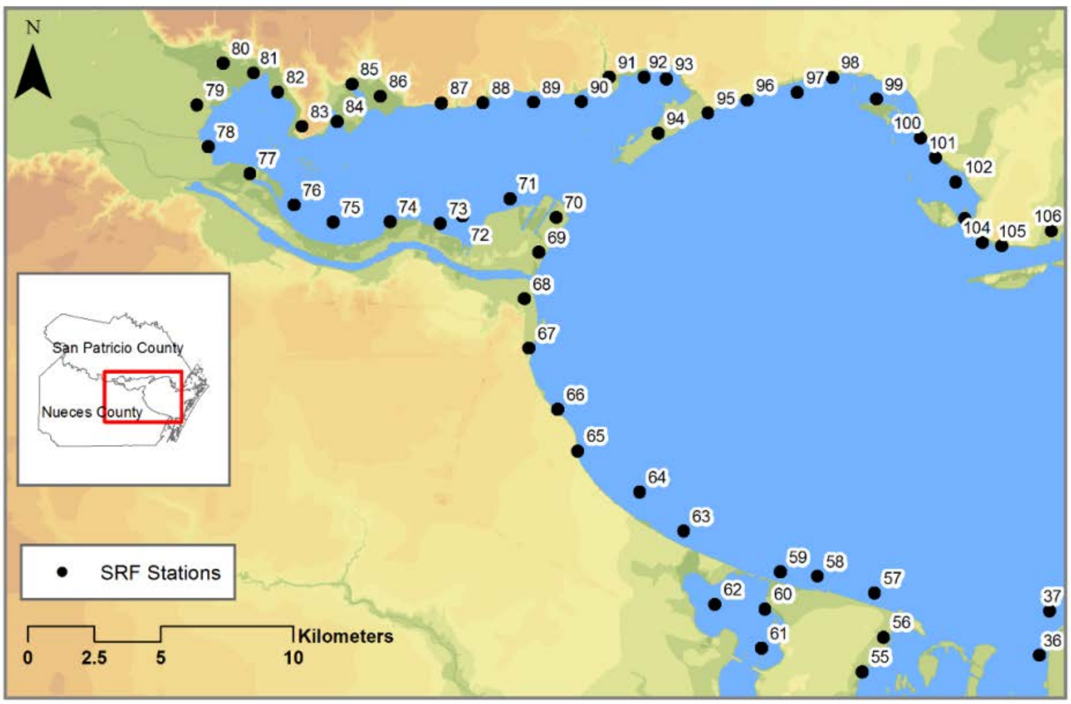


Figure 2-1 Example SRF Stations

Table 2–1 Example of Surge Response Data at SRF Stations

c_p (mb)	R_p (nm)	v_f (m/s)	θ (°)	x_0 (km)	Surge(m) at SRF station 61	Surge(m) at SRF station 62	Surge(m) at SRF station 63	Surge(m) at SRF station 64
770	4	3	-80	-80	1.4931	1.4312	1.7854	1.7345
770	4	3	-80	-75	1.5196	1.457	1.8181	1.7664
770	4	3	-80	-70	1.5471	1.4838	1.8519	1.7995
770	4	3	-80	-65	1.5756	1.5115	1.887	1.8339
770	4	3	-80	-60	1.6051	1.5403	1.9234	1.8696
770	4	3	-80	-55	1.6358	1.5702	1.9613	1.9068
770	4	3	-80	-50	1.6676	1.6013	2.0007	1.9454
770	4	3	-80	-45	1.7008	1.6336	2.0418	1.9856
770	4	3	-80	-40	1.7352	1.6673	2.0845	2.0276
770	4	3	-80	-35	1.7711	1.7024	2.1291	2.0713
(continue)	…	…	…	…	…	…	…	…

Note. c_p: Hurricane central pressure; R_p: Hurricane radius to the maximum wind; v_f: Hurricane forward speed; θ: Hurricane approach angle with respect to the shoreline.

JPM was developed in the 1970s (Ho and Myers, 1975; Myers, 1975) to evaluate hurricane surge frequency. Recent implementations of JPM have involved surge response approach to obtain the surge with respect to hurricanes with parameters which are not in the historical records (Resio et al., 2009; Niedoroda et al., 2010). The JPM implementations combined with surge-response approach need to consider a large number of combinations of hurricane parameters, and each of such combination requires the simulation of wind, waves and surge (Niedoroda et al., 2008), hence the simulations of surges usually require huge amount of computing resources. Resio et al. (2009) emphasize the need to develop a method in which only a smaller optimal sample of storms needed to be simulated. Toro et al. (2010) and Niedoroda et al. (2008) developed two optimization techniques to identify the smaller optimal sample, namely response-surface JPM-OS scheme (JPM-OS-RS) and quadrature JPM-OS scheme (JPM-OS-Q). The quadrature JPM-OS scheme uses an algorithm that optimally selects the parameters combinations and assigns an appropriate weight to each synthetic storm, whereas the response-surface JPM-OS scheme carefully selects a set of synthetic storms and interpolates between the surge results among the selected storms (Toro et al., 2010).

The hurricane parameters considered in JPM includes hurricane central pressure (c_p), radius to the maximum wind (R_p), forward speed (v_f), approach angle (θ), and landfall location (x_0). The probability density functions of these hurricane parameters are estimated based on analysis of the historical Gulf of Mexico (GOM) hurricane records (HURDAT; Landsea et al., 2013) and used as present-day conditional probability density functions (PDFs). Irish and Resio (2013) assume the conditional probability of central pressure on landfall location follows a Gumbel distribution (Eq.(2.3)); whereas the conditional probabilities of hurricane radius to the maximum wind on central pressure, forward speed on approach angle, and approach angle on landfall location follows a normal distribution (Eq. (2.4) through (2.8)).

$$p(c_p, R_p, v_f, \theta, x_0) = \Lambda_1\Lambda_2\Lambda_3\Lambda_4\Lambda_5 \tag{2.3}$$

$$\Lambda_1 = p(c_p|x_0) = \frac{1}{a_1(x_0)} \exp\left[\frac{\Delta p - a_0(x_0)}{a_1(x_0)}\right] \exp\left\{- \exp\left[\frac{\Delta \mathrm{p} - \mathrm{a}_0(x_0)}{a_1(x_0)}\right]\right\} (Gumbel\ Distribution) \tag{2.4}$$

$$\Lambda_2 = p(R_p|c_p) = \frac{1}{\sigma(\Delta p)\sqrt{2\pi}} \left\{- \exp\left[\frac{\left(\bar{R}_p(\Delta p) - R_p\right)^2}{2\sigma^2(\Delta p)}\right]\right\} (Normal\ Distribution) \tag{2.5}$$

$$\Lambda_3 = p(v_f|\theta) = \frac{1}{\sigma(\theta)\sqrt{2\pi}} \left\{- \exp\left[\frac{\left(\bar{v}_f(\theta) - v_f\right)^2}{2\sigma^2(\theta)}\right]\right\} (Normal\ Distribution) \tag{2.6}$$

$$\Lambda_4 = p(\theta|x_0) = \frac{1}{\sigma(x_0)\sqrt{2\pi}} \left\{- \exp\left[\frac{\left(\bar{\theta}_f(x_0) - \theta\right)^2}{2\sigma^2(x_0)}\right]\right\} (Normal\ Distribution) \tag{2.7}$$

$$\Lambda_5 = g(x_0) \tag{2.8}$$

where

Λ_i: *The probability density function for each considered parameters*

Δp: *The central pressure deficit (peripheral pressure minus c_p)*

a_i: *The Gumbel coefficients*

σ: *The standard deviations of the normal distribution*

overbars: Normal distribution mean values

$p(c_p|x_0)$: *The conditional probability of central pressure given landfall location*

$p(R_p|c_p)$: *The conditional probability of radius to maximum wind given central pressure*

$p(v_f|\theta)$: *The conditional probability of forward speed given approach angle*

$p(\theta|x_0)$: *The conditional probability of approach angle given landfall location*

$g(x_0)$: *The rate of hurricane landfall occurance per unit coastal length*

3. METHODOLOGY

The overall goal of this study is to construct a framework that can be used to: (1) develop expected annual loss map at parcel and census block level; (2) develop flood damage (also affected population, flooded area) vs. return period relationship for parcel data; (3) develop affected business (including affected number of business, employee, and sales volume) vs. return period relationship for business data; (4) estimate the surge flood damage for given hurricane scenarios; and (5) develop flood maps for hurricane scenarios of interest. Using this framework, hurricane surge flood damage for current and future climate conditions (i.e., B1, A1B, A1FI in year 2030 and 2080) will be evaluated and, by comparison, the hurricane surge induced damage will be estimated for three selected cities, namely Corpus Christi, TX, Gulfport, MS and Panama, FL.

3.1. Mapping Expected Annual Parcel Losses Caused by Hurricane Surge Flooding

3.1.1. Individual Parcel Loss Caused by a Hurricane Surge Flood

Surge flood losses for a specific parcel are evaluated with damage curves, which provide the percent damage of the parcel improvement (i.e., built structure) for a given flood depth and building type (see Figure 3-1). The legend in Figure 3-1 shows the building type. For example, RES1 stands for one-floor, two-floors, three-floors or split-level residential structure with or without basement; RES2 stands for mobile home. In HAZUS database there are damage curves for costal A zone and costal V zone (see Appendix A and B for the curve descriptions and Appendix C for the graphs of these curves). V-Zones are the areas of the coastal floodplain which is subject to wave height more than 3ft, whereas A-Zones are areas subject to wave height less than 3 ft (FEMA, 2014). For the building type with multiple curves, such as residential 1, we assumed the buildings at the study areas are all with one floor and without basement, for that the buildings at water-prone area are less likely to have basement. Then we average the value of the left curves to be the representative curve for each building type, as is shown in Figure 3-1. The flood depth was evaluated as the surge water elevation minus the elevation of the first floor of the building.

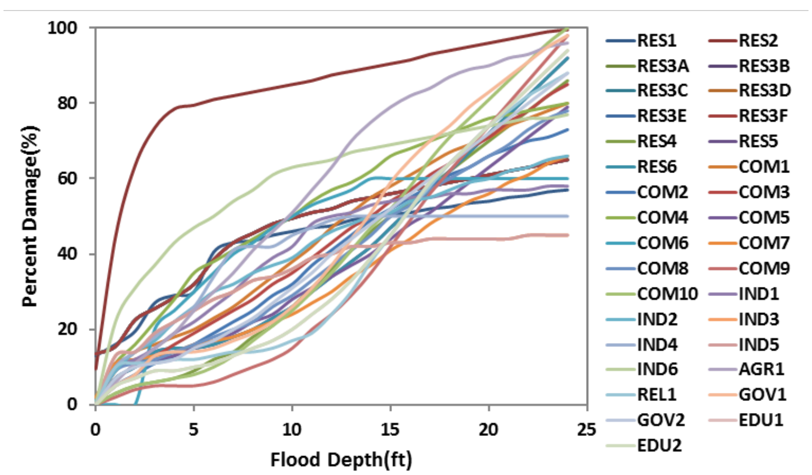


Figure 3-1 Example of Damage Curves for Different Types of Buildings

The surge water elevation at a parcel is complex to estimate because it depends on the event, on the affected area and on the geographic location of the parcel within the affected area. In fact, the water does not rise horizontally as a bath tub but, because of the heterogeneities in the hurricane wind field, surface topography, ocean bathymetry and shoreline shape, it exhibits a different elevation in each location. Here, however, we assumed that the surge water elevation on a shore point could be projected horizontally inland within an area of influence and, for that purpose, a number of points along the shoreline were defined. For reasons that will become clear below, these points were called SRF stations (where SRF, as spelled-out above, stands for Surge Response Function). For each SRF station, the area of influence was delineated as the drainage area of the shoreline segment between the point being considered and its two neighboring points (Figure 3-2) (See Appendix E for the process of generating SRF zones using ArcGIS). For the convenience, the areas of influence are called SRF zones, as is they correspond to SRF points. The identification of these SRF zones is also complex and the approach presented here applies when the surge is low enough to not overtop the drainage divides. Identifying the exact boundary between these SRF zones, though, should not be of critical importance since the SRF points should be close enough to each other that no significant differences in water elevation between two consecutive points should exist. The surge elevations at the SRF stations were determined with surge response functions, in which the independent variables are hurricane variables, including central pressure c_p, radius to the maximum wind R_p, forward speed v_f, approach angle θ, and landfall location x_0.

Figure 3-2 Example of SRF Stations and Corresponding SRF Zones

Because the information of the first floor elevation for these study areas is not available, we adopt Frey et al. (2010) assumption that all the buildings in this study have first floor elevation of one foot above the ground level given by the Digital Elevation Model (DEM). Although this assumption could be considered questionable, it was not possible to use the actual values because they change from parcel to parcel.

The building type was not possible to infer from the parcel data available. However, the HAZUS database includes the number of buildings of each type in each census block. Therefore, even if it was not possible to determine the building type, the chance that it would fall under one type or another was known. To account for not knowing the parcel building type, census-block damage curves were developed as a weighted average of the damage curves of the different building types in the block. That is

$$\overline{D_J}(d) = \sum_i f_{ij}\, D_i(d) \tag{3.1}$$

where fij is the fraction of building type i in block j, d is the water depth, $\overline{D_J}(d)$ is the percent damage for building type i and water depth d, and $\overline{D_J}(d)$ is the weighted averaged percent damage curve for block j (see Figure 3-3). Note that $\sum_i f_{ij}$ for all blocks j.

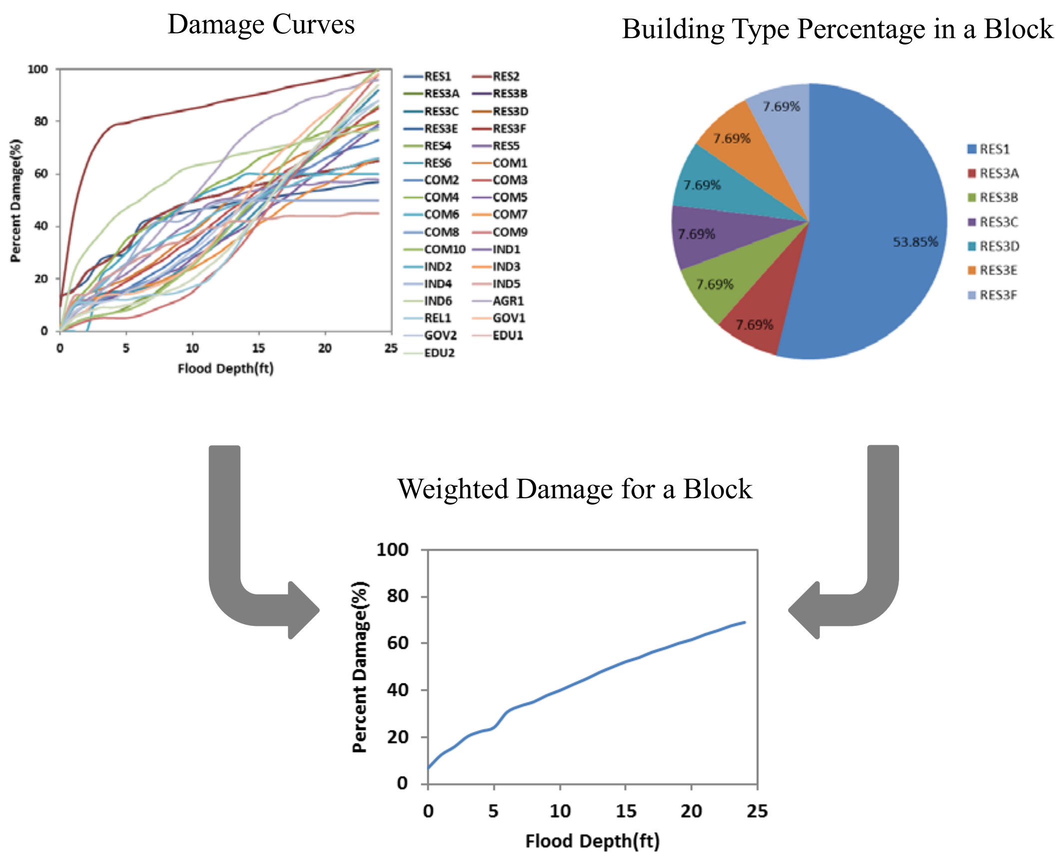


Figure 3-3 Example of Obtaining the Weighted Damage Curve in A Block

The losses were equal to the product of the percent damage given by the damage curve for the structure type times the improvement value, and can be expressed as

$$L_{km} = \overline{D}_j(d_{km}) \times V_k \tag{3.2}$$

where Lkm are the losses in parcel k of block j caused by hurricane m, dkm is the water depth in parcel k caused by hurricane m, and Vk is the improvement value in parcel k. Note that the improvement value does not include the land value since it is assumed that the land is not affected by flooding. Other value losses, not directly related to the structure damage, are not being considered in this analysis.

3.1.2. Individual Parcel Risk Caused Hurricane-Surge Flood

To calculate the hurricane-surge flood risk or *expected annual losses*, the damage caused by individual hurricane surges and their probability of occurrence should be known. The flood risk, for an individual parcel *k* located in block *j*, can be expressed as

$$R_k = \sum_m L_{km} \times p_m \tag{3.3}$$

where R_k is the flood risk for parcel *k* located in block *j*, and p_m is the probability of occurrence of hurricane *m* in a given year. Note that $\sum_i p_m = \lambda$, where λ is the probability that hurricanes would make landfall on a specified coastline range. Based on the dataset in Irish et al. (2011b), λ is approximately 0.36 for a 2,000 m coastline along Gulf of Mexico.

Determination of the water depth in a parcel for all likely hurricanes required the determination of surge water elevations at its SRF point for all likely hurricanes, which in turn, are required of hydrodynamic simulation. These individual simulations are very computer intensive and, in the case of needing simulations for a large number of hurricanes, it could be almost impossible to conduct. To address this problem, SRFs were developed to estimate the surge elevation at the SRF points for a number of hurricanes resulting from the combination of the five hurricane parameters listed above. The probability of occurrence of each hurricane depended on its parameters and their probability distributions, and was determined with the JPM. It is understood that each individual hurricane will have a different probability of occurrence and that, all together, represent the hurricane surge distribution for the corresponding SRF point.

3.1.3. Aggregated Risk Caused by Hurricane-Surge Flood

The hurricane-surge flood risk at a census block level, city level or at any other aggregation level, can be estimated as the sum of the corresponding parcel risks. The aggregation of the parcel risks – which are expected damage values (i.e., first moments of the damage) – is justified because the first moment of the sum of variables is equal to the sum of their first moments regardless the variables are or are not independent; that is, the risk of a set of parcels, is equal to the sum or their risks. It is expressed as

$$R = \sum_{k} R_k \tag{3.4}$$

Once the risks of any individual parcel or at any aggregation level have been determined, risk maps can be developed. These risk maps can represent expected loss values in dollars or in percentage of the parcel improvement value (see Figure 3-4 and Figure 3-5).

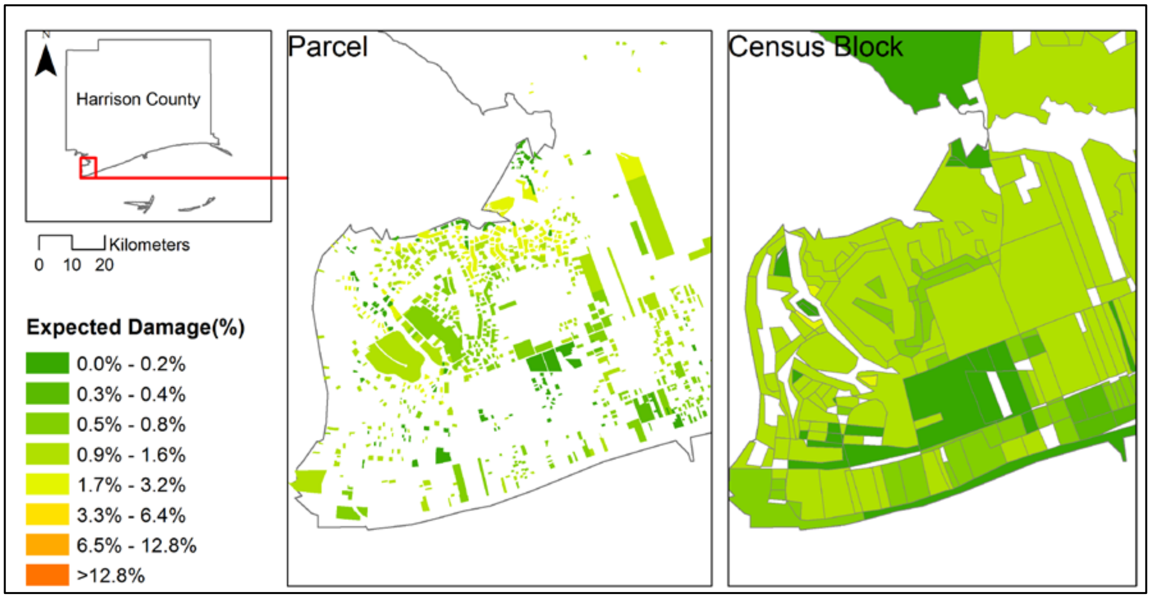


Figure 3-4 Example of Expected Damage in Percentage of Parcel Improvement Value

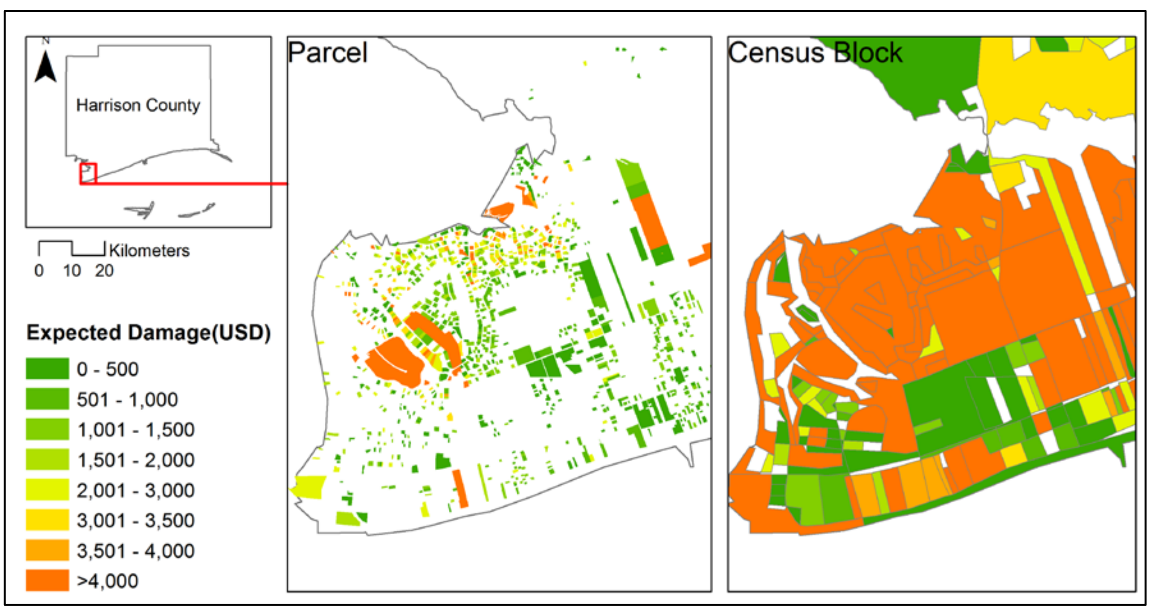


Figure 3-5 Example of Expected Damage in Dollars

3.2. Damage, Affected Population, and Flooded Area for Different Hurricane Scenarios and Return Periods

To estimate the surge flood damage, population affected, and flooded area by hurricane surge flooding, we apply two approaches. One is a "hurricane scenario" approach, which yields the damage statistics for given hurricane characteristics; and the other is a "return-period" approach, which gives expected values for given return period. As is mentioned before, the flood damage at parcel level can be evaluated with Eq.(3.2).

3.2.1. The Hurricane Scenario Approach

As is discussed in the background section, for each hurricane scenario, surge values at SRF can be determined by given hurricane parameters. For estimating the damage caused by flooding, the surge at the coast is projected horizontally inland within the corresponding SRF zones. The population affected is estimated assuming that the population in the census blocks is evenly distributed over the parcel areas. People in a parcel are considered affected if the parcel is totally or partially flooded, regardless of the water depth. The number of people affected at any aggregation level, such as census block or city, is obtained as the sum of the people affected at the parcel level. On the other hand, the flooded areas are identified as those areas in which the surge elevation is greater than the ground elevation given by the DEM. The surge elevation at a given point depends on the hurricane event as well as on the SRF stations and SRF zones within which the point is located. The plots below (Figure 3-6) show damage, population affected and flooded area for given hurricane parameters (In this example, c_p=950 mb, R_p=16nm) and different landfall locations. Note that the numbers of landfall locations in this case are specified for the coastline for Harrison County, MS (see Figure 3-7).

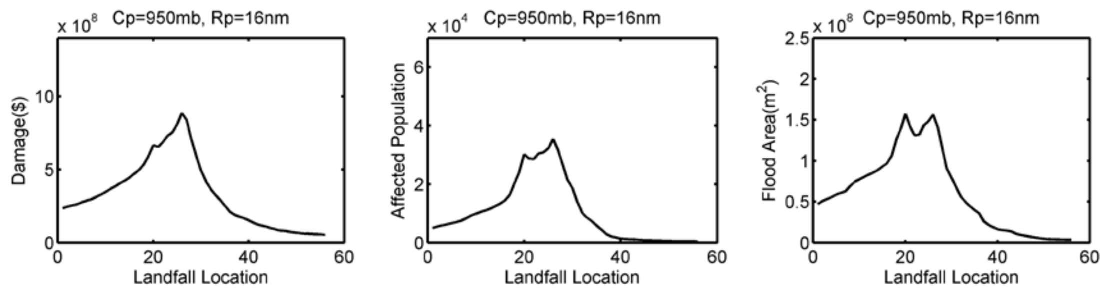


Figure 3-6 Example Results of Hurricane Scenario Approach Analysis

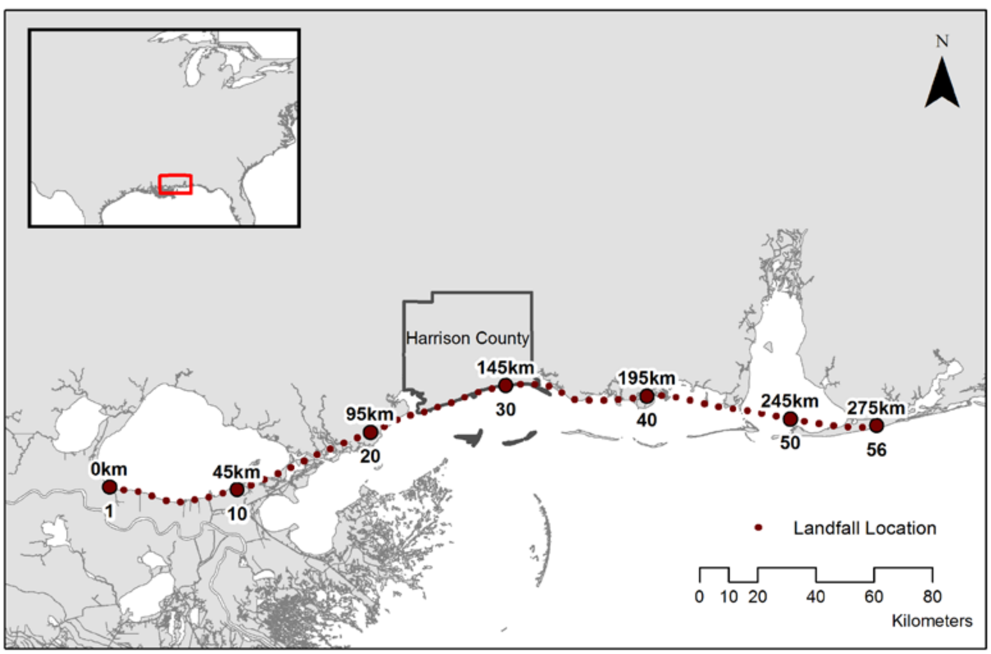


Figure 3-7 Gulfport and Harrison County, MS Landfall Locations

3.2.2. The Return-Period Approach

If we denote the surge response for a given hurricane as ζ, then the probability of that surge value occurring as a result of the hurricane can be represented as:

$$p(\zeta) = p_c\left(c_p, R_p, v_f, \theta, x_0\right) \tag{3.5}$$

where p_c represents "continuous" distribution of the joint probability of hurricane parameters. Consider the distribution in discrete format, Eq.(3.5) can be written as:

$$p_{ijklm}(\zeta) = p_d\left(c_p, R_p, v_f, \theta, x_0\right) \tag{3.6}$$

where the subscripts ijklm are the indices of the five parameters, p_d represents "descrete" distribution of the joint probability of hurricane parameters. Thus the cumulative distribution function for the surge response x gives:

$$F(x) = \sum p_{ijklm}(\zeta < x) \tag{3.7}$$

Then the return period of surge value x is defined by

$$T(x) = \frac{1}{\lambda\left[1 - F(x)\right]} \tag{3.8}$$

Where $T(x)$ is the return period for surge value x. λ is the probability that hurricanes would make landfall on a specified coastline range. In this study, we extend the surge-return period concept to damage-return period used in Irish et al. (2012) and Irish and Resio (2013). Following the same rationale, the damage value D (either flood damage, population affected, or flooded area) as a result of the hurricane can be represented as:

$$p_{ijklm}\left(d\right)=p_{d}\left(c_{p},R_{\max},v_{f},\theta,x\right) \tag{3.9}$$

Hence the cumulative distribution function for the damage and damage-return period can be represented as:

$$F\left(d\right)=\sum p_{ijklm}\left(D<d\right) \tag{3.10}$$

$$T\left(d\right)=\frac{1}{\lambda\left[1-F\left(d\right)\right]} \tag{3.11}$$

The results of the return period analysis give the long term risk of hurricane surges on the damage statistics such as flood damage, affected population, and flood area (Figure 3-8). Instead of obtaining the surge-return period relations at SRF points. The return-period approach presented here gives the risk of damage statistics for the entire area of interest.

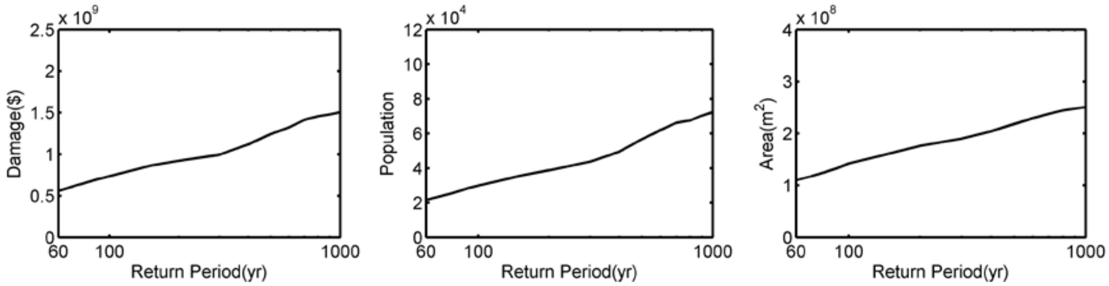


Figure 3-8 Example Results of Return Period Approach Analysis

3.3. Affected Businesses

We also analyze the effects of hurricane surge flooding on businesses. The business data is obtained from ReferenceUSA (Infogroup, 2012). The ReferenceUSA database holds business information such as company name, company address, employee size and sales volume, etc. (see Table 3–1 for list of fields). Figure 3-9 shows an example of the business data at Corpus Christi, TX. For business data, we conducted damage analysis using the Return Period Approach and chose employee size, sales volume and affected business as the variables for showing the damage on business due to hurricane surge flooding. Similar to the analysis of parcel data, this study assumed that as the water elevation gets above the ground elevation of a business data point, this business is consider affected and the employee number and sales volume of this business are also considered as affected.

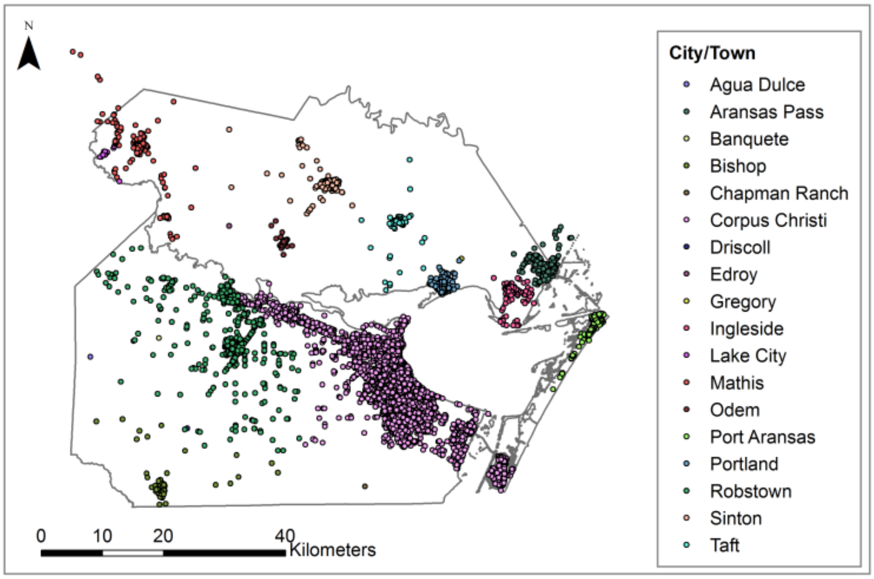


Figure 3-9 Corpus Christi, TX Business Data

Table 3–1 Reference USA Business Data Fields

No	**Business Data Field**
1	Company Name
2	Executive First Name
3	Executive Last Name
4	Address
5	City
6	State
7	ZIP Code
8	Credit Score Alpha
9	Executive Gender
10	Executive Title
11	Fax Number Combined
12	IUSA Number
13	Location Employee Size Actual
14	Location Employee Size Range
15	Location Sales Volume Range
16	Location Sales Volume Actual
17	Phone Number Combined
18	Primary SIC Code
19	Primary SIC Description
20	SIC Code 2
21	SIC Code 2 Description
22	Record Type
23	Primary NAICS
24	Primary NAICS Description
25	Latitude
26	Longitude

3.4. Climate Change Effect

The effects of climate change on hurricane flood damage are considered by taking into account sea level rise (SLR) and hurricane intensification due to the sea surface temperature (SST) increase. To analyze theses effects, we use three future climate scenarios from Intergovernmental Panel On Climate Change report (IPCC, 2007a), and the documented SST/SLR values (Irish et al., 2008a; National Research Council, 2012) that correspond to these scenarios. Calculations were conducted for year 2030 and 2080 for our case study. The climate scenarios selected were B1, A1B and A1FI. In IPCC's definition, scenario B1 has a global mean temperature rise estimate with a low rate of global warming, A1B has a global mean temperature rise estimate with a mid-ranged global warming, and A1FI has a global mean temperature rise estimate with a high rate of global warming. It was considered that these three scenarios covered the entire spectrum of possible future climate conditions. The SLR and SST values correspond to these selected scenarios and the value selection rationale is discussed in 3.4.3.

3.4.1. Consideration of Sea Level Rise (SLR)

Sea level rise in the future will cause inundation of coastal areas and, hence, affect the population and built environment. According to NOAA (2014), the global sea level has been climbing in the 20th century, and the observation records show that the rising rate has increased in recent decades. At current time, the worldwide SLR rate is approximately 3 mm/yr. As SLR could affect the population and economic activities in coastal area, many studies have focused on the impact of SLR on hurricane prone coastal areas (Wu et al., 2002; Cooper et al., 2008; Frazier et al., 2010; Bjarnadottir et al., 2011; Shepard et al., 2012).

Future SLR is expected to flood land and current-day property; however, it is expected that the population affected will relocate before SLR flooding occurs. In this study, we calculate the expected damage with respect to different return period for selected SLR scenarios and will consider the properties and population below the sea level of each scenario as complete loss. That is, we assume no population relocation and assume once a parcel is below the SLR, the entire population is affected and the total property of the parcel is lost.

Resio and Westerink (2008) state that the storm surge is affected by water depth and shelf width. Other studies also reported the effect of SLR on storm surge (Smith et al., 2010; Mousavi et al., 2011). That is, the effect of SLR on surge cannot be evaluated as linearly adding the amount of SLR on simulated surge value. Udoh (2012) investigated the relationship between SLR and surge for a case study of Corpus Christi, he found the relationship to be Eq. (3.12). In this study the surge values with SLR effects will also be adjusted by Eq. (3.12).

$$\zeta_s = \left(\eta_s - SLR\right) = k\zeta_0 + l \tag{3.12}$$

where

$$\begin{cases} \zeta_0 : \textit{the current day surge value} \left(\textit{without effect of } SLR\right) \\ \zeta_s : \textit{the projected surge values with effect of } SLR \\ \eta_s : \textit{the water elevation with effect of } SLR \\ k, l : \textit{adjustment coefficients for surge value} \end{cases}$$

For Gulfport and Panama City, Eq.(3.12) has been revised as Eq.(3.13), which determines the coefficients k and l by fitting the surge value with SLR effect (ζ_s) and the current surge plus SLR ($\zeta_o + SLR$).

$$\zeta_s = k\left(\zeta_0 + SLR\right) + l \tag{3.13}$$

3.4.2. Consideration for Sea Surface Temperature (SST) Rise

According to Knutson and Tuleya (2004), the intensification, in terms of central pressure decrease, alters the probability distribution of hurricane central pressure c_p and hence changes the entire joint probability of hurricane parameters $p(c_p, R_p, v_f, \theta, x_0)$ and modified the calculations of damage, population affected and flooded area. Increasing sea surface temperature (SST) is recognized as a result of increased CO^2 concentration in the atmosphere (IPCC, 2007b). Studies show that increased SST will result in stronger hurricane (Emanuel, 2005; Elsner, 2006; Elsner et al., 2008), hence cause more severe damage. Knutson and Tuleya (2004) estimated that on average the hurricane intensity would increase 8% for every 1^{o}C of SST rise. In this study we adopt Knutson's assumption and apply Eq.(3.14) to adjust the hurricane central pressure probability distribution.

$$p_{\Delta SST} = p_0 - 0.08(\Delta SST)\Delta p \tag{3.14}$$

$$\Delta p = p_{atm} - p_0 \tag{3.15}$$

where

$$\begin{cases} p_{\Delta SST} : \mathit{future\ projected\ hurricane\ central\ presure} \\ p_0 : \mathit{current\ day}\,(2000s)\ \mathit{hurricane\ central\ pressure} \\ \Delta SST : \mathit{sea\ surface\ temperature\ change} \\ \Delta p : \mathit{the\ central\ pressure\ deficit}\,(\mathit{peripheral\ pressure\ minus}\ p_0) \end{cases}$$

As mentioned in the background section, the hurricane central pressure given landfall location follows the Gumbel distribution. In this study, it is assumed that the central pressure is independent of landfall location, hence Eq. (2.4) is simplified as:

$$\Lambda_1 = p(c_p) = \frac{1}{a_1}\exp\left[\frac{\Delta p - a_0}{a_1}\right]\exp\left\{-\exp\left[\frac{\Delta \mathrm{p} - \mathrm{a}_0}{a_1}\right]\right\}(\mathit{Gumbel\ Distribution}) \tag{3.16}$$

Where the coefficients a_0 and a_1 are obtained by fitting the distribution with current day hurricane central pressure data from The North Atlantic hurricane database (HURDAT, 2014). For the situations that consider SST, we obtain the new hurricane central pressure by implementing Eq. (3.13) and (3.14), then fit the Gumbel distribution again to retrieve the new coefficients that correspond to specified SST. Figure 3-10 shows the shift of hurricane central pressure distribution due to SST.

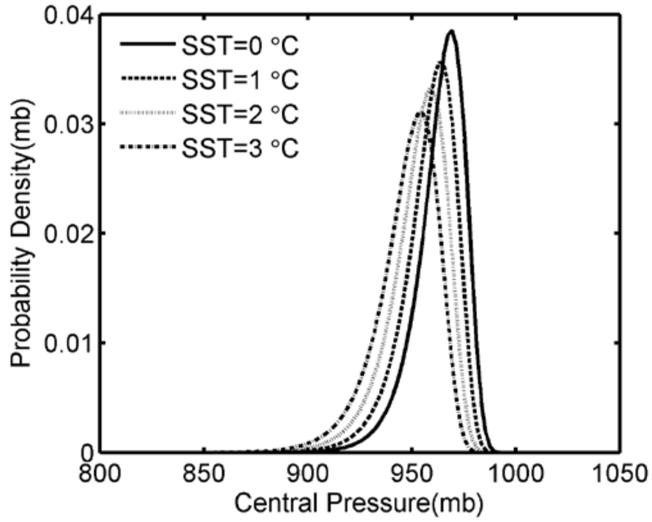


Figure 3-10 Shift of Central Pressure Deficit

3.4.3. SLR and SST Values Selected for Conducting Analysis

According to (Irish et al., 2008a), the SST rise projections range from 0.36 to 1.38 ℃ for the 2030s and from 0.96 to 5.02 ℃ for the 2080s, depending on assumptions regarding global warming and model sensitivity parameter (See Table 3–3, note the temperature sensitivity is the sensitivity for CO^2 doubling, which is the change of temperature in ℃ associated with a doubling of the CO^2 concentration.). Here for B1 scenario in 2030s we choose the SST of "cool" scenario with 2℃ sensitivity, which is 0.507℃ (see Table 3–3, the lowest one for B1 scenario and 2℃ sensitivity) to represent the best scenario; for A1B scenario in 2030s we choose the SST of "average" scenario with 3℃ sensitivity, which is 0.848℃ (see Table 3–3, the medium one for A1B scenario and 3℃ sensitivity) to represent the mid-range scenario; for A1FI scenario in 2030s we choose the SST of "warm" scenario with 4.5℃ sensitivity, which is 1.233℃ (see Table

3–3, the highest one for A1FI scenario and 4.5℃ sensitivity) to represent the worst scenario. Similarly, we choose 0.958 ℃, 2.493 °C and 5.204 ℃ as the SSTs for B1, A1B and A1FI scenarios 2080s, respectively. The SST values for selected scenarios and time is listed in Table 3–4.

For the SLR values of the selected scenarios, we adopt the global sea level rise in (National Research Council, 2012) with consideration of local SLR trend from NOAA sea level gages and Church et al. (2004) (see Table 3–2). Take the SLR value of Corpus Christi at 2030 under A1B scenario, as an example, the SLR value is adding the global SLR value at 2030 (Figure 3-11) on the local SLR, which is subtracting the NOAA gage estimated trend by the global SLR trend (Church et al., 2004). Hence the SLR for Corpus Christi at 2030, under A1B scenario is $13.5\,cm + (5.16\,mm/yr - 1.8\,mm/yr) \times 30\,yr = 24\,cm$. Similarly, the SLR value of Gulfport at 2030, under A1B scenario, is obtained by adding the global SLR value at 2030 on the local SLR, which is subtracting the NOAA gage estimated trend by the global SLR trend (Church et al., 2004). Hence the SLR value for Gulfport at 2030, under A1B scenario, is $13.5\,cm + [0.5 \times (9.24\,mm/yr + 2.98\,mm/yr) - 1.8\,mm/yr] \times 30\,yr = 17\,cm$. For Panama City, the SLR value of Panama City at 2030 under A1B scenario, is obtained by adding the global SLR value at 2030 on the local SLR, which is subtracting the NOAA gage estimated trend by the global SLR trend (Church et al., 2004). Hence the SLR value for Panama City at 2030 under A1B scenario, is $13.5\,cm + (0.75\,mm/yr - 1.8\,mm/yr) \times 30\,yr = 17\,cm$. Applying the same rationale, the SLR values for all selected scenarios and time are listed in Table 3–4.

Table 3–2 Estimate of Local SLR Rate Contribution at Corpus Christi, Gulfport and Panama City

Corpus Christi, TX:

- Rockport, TX NOAA gauged SLR: 5.16 mm/yr ± 0.67 mm/yr (1948-2006)
- Observed global SLR: 1.8 mm/yr ± 0.3 mm/yr (observed global, Church et al. (2004) [1950-2000])
- Local contribution = 5.16 - 1.8 ≅ 3.4 mm/yr

Guflport, MS:

- Grand Isle, LA NOAA gauged SLR: 9.24 mm/yr ± 0.29 mm/yr (1947-2006)
- Dauphin Island, AL NOAA gauged SLR: 2.98 mm/yr ± 0.87 mm/yr (1966-2006)
- Assume average is reprehensive, estimated Gulfport rate is ~=6.11 mm/yr
- Observed global SLR: 1.8 mm/yr ± 0.3 mm/yr (observed global, Church et al. (2004) [1950-2000])
- Local contribution = 6.11 - 1.8 ≅ 4.3 mm/yr

Panama, FL:

- Panama City, FL NOAA gauged SLR: 0.75 mm/yr ± 0.83 mm/yr (1948-2006)
- Observed global SLR: 1.8 mm/yr ± 0.3 mm/yr (observed global, Church et al. (2004) [1950-2000])
- Local contribution = 0.75 - 1.8 ≅ -1.1 mm/yr (local contribution gives sea-level lowering)

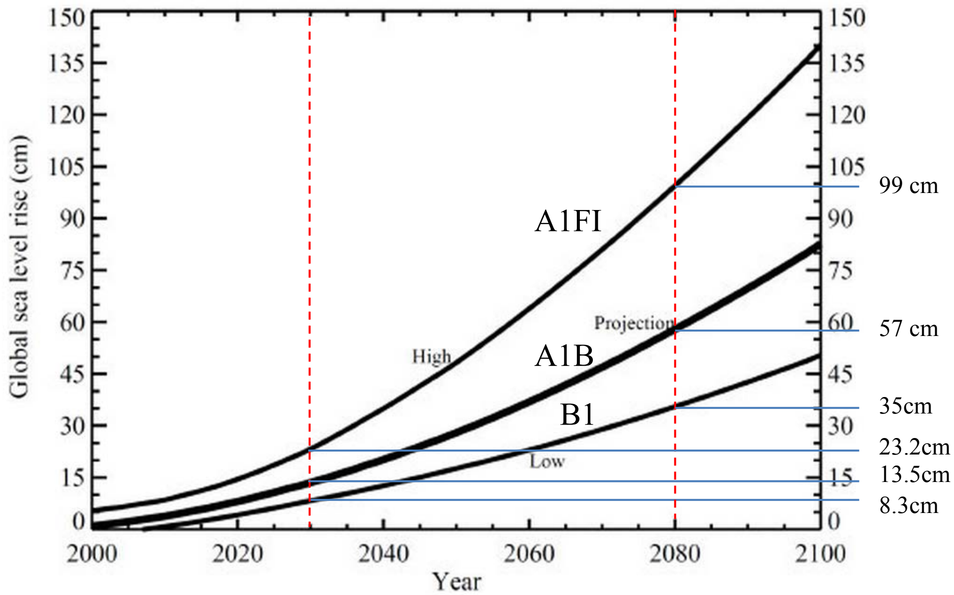


Figure 3-11 Global Sea Level Rise Projection (Table5.5 in National Research Council (2012) p.93)

Table 3–3 Projected 6-Month Average Sea Surface Temperature (SST) Warming Using MAGICC/SCENGEN (Table 2 in Irish et al. (2008a))

Climate scenario B1 projected SST rise(°C)			**Climate scenario A1B projected SST rise(℃)**			**Climate scenario A1FI projected SST rise(°C)**		
Cool	Average	Warm	Cool	Average	Warm	Cool	Average	Warm
				2030s (2 ℃ sensitivity)				
0.507	0.718	0.84	0.399	0.645	0.803	0.357	0.609	0.755
				2030s (3 ℃ sensitivity)				
0.664	0.940	1.101	0.516	0.848	1.041	0.469	0.795	0.985
				2030s (4.5 ℃ sensitivity)				
0.832	1.180	1.381	0.640	1.055	1.296	0.589	0.996	1.233
				2080s (2 ℃ sensitivity)				
0.958	1.839	2.155	1.715	2.493	2.944	2.166	3.298	3.955
				2080s (3 sensitivity)				
1.294	1.839	2.155	1.715	2.493	2.944	2.166	3.298	3.955
				2080s (4.5 ℃ sensitivity)				
1.678	2.394	2.81	2.202	3.211	3.796	2.745	4.188	5.024

Table 3–4 SLR/SST Values Used For Selected Climate Scenarios

Climate Scenario[1]	ΔSST(°C)	SLR(m) global	SLR(m) Corpus Christi, TX	SLR(m) Gulfport, MS	SLR(m) Panama, FL
Cur2000*	0	0	0	0	0
B130Low	0.51	0.08	0.19	0.12	0.05
A1B30Mid	0.85	0.14	0.24	0.17	0.10
A1FI30High	1.23	0.23	0.33	0.27	0.20
B180Low	0.96	0.35	0.62	0.47	0.26
A1B80Mid	2.49	0.57	0.84	0.67	0.48
A1FI80High	5.02	0.99	1.26	1.09	0.90

* Cur2000: current day (2000) climate condition

3.5. Developing Flood Maps

In the Background section, we have discussed that the surge values that correspond to a hurricane event can be obtained from surge response functions. Also we have made the assumption that within a SRF zone, the surge value is a constant. Thus we can overlay the surge values and DEM to determine the flooded area as where the surge value is greater than the ground elevation. The rationale of flood map developing process is shown in Figure 3-12. Figure 3-13 is an example of flood map of Corpus Christi, given the following hurricane parameters: $c_p = 950\, mb, R_p = 16\, nm, v_f = 3\, knots, \theta = 0^o, x_0 = 30km.$ Note the hurricane symbol denotes the hurricane center and the red radius of the red circle is the radius to the maximum. (16 nm ≅ 30km). In sum, the surge response function allows us to efficiently generate flood maps that correspond to different hurricane events, since we do not need to run the hydrodynamic model for all

[1] Note. The number after scenario stands for the future time point. E.g. B130 means B1 scenario at year 2030; "Cur2000" means current condition.

the hurricane scenarios of interest. One application is to compare the flood maps of hurricanes with the same size and strength but make landfall at different locations. In section 5.5 we will develop the flood maps for a hypothetical hurricane that makes land at the three study areas and compare the results.

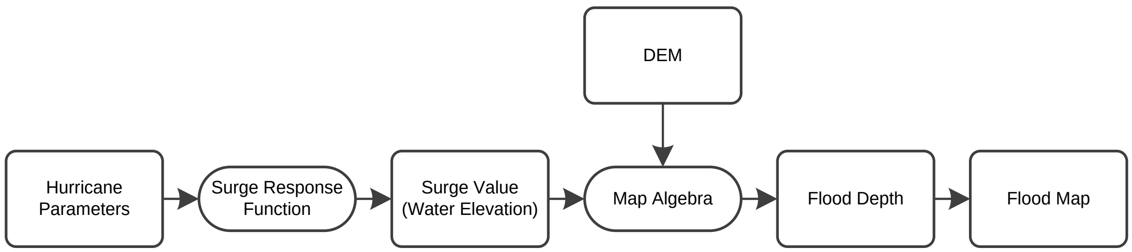


Figure 3-12 Flood Map Developing Process

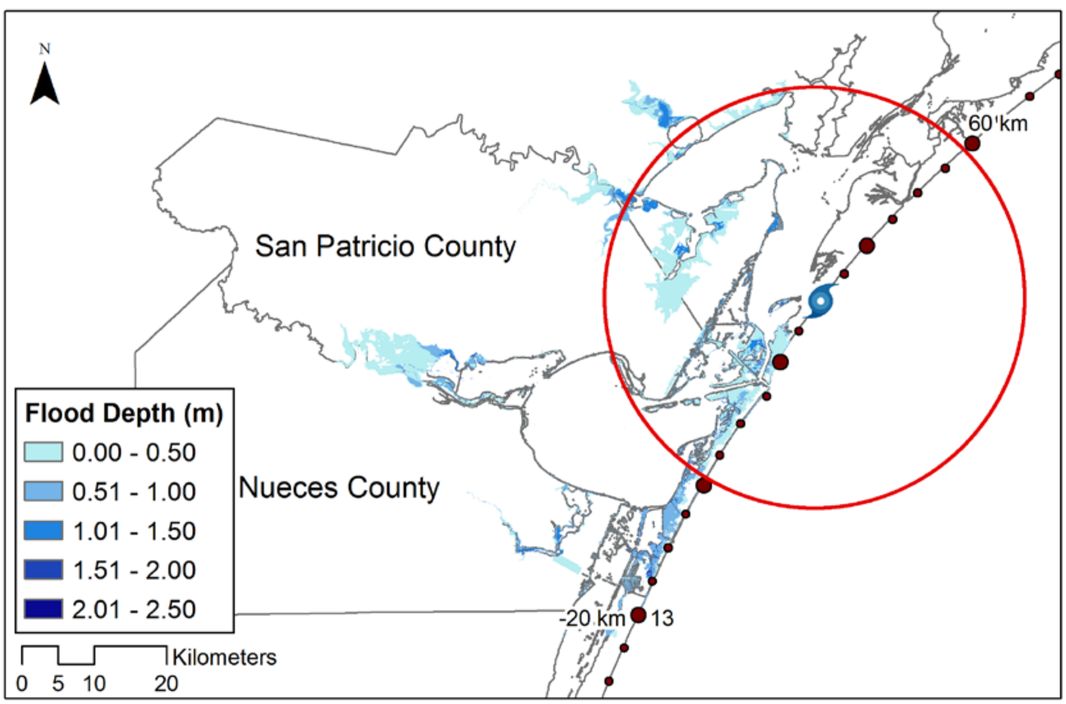


Figure 3-13 Example Flood Map

4. STUDY AREAS AND DATA

4.1. Study Areas

Three study areas were considered: Corpus Christi, TX and Gulfport, MS (Figure 4-1) on the Gulf of Mexico coastline, and subject to hurricanes. Corpus Christi has faced Hurricane Beulah in 1967, Hurricane Bret in 1999 and Hurricane Alex in 2010. Gulfport has faced Camille in 1969, Ivan in 2004 and most notable recently, Hurricane Katrina in 2005. Our flood damage analysis will include Nueces and San Patricio Counties for the case of Corpus Christi City, Harrison County for the case of Gulfport City and Bay County for Panama City. These counties were selected to include, not only the areas within the city limits but also the surrounding metropolitan area.

Figure 4-1 Three Study Areas along Gulf of Mexico Coast (Basemap Image Source: ESRI)

Corpus Christi City is located on the south coast of Texas (Figure 4-2). Nueces County and San Patricio County have a population of 413,291, with 347,691 in Nueces County (including 312,195 in Corpus Christi) and 65,600 in San Patricio County (U.S. Census Bureau, 2013c).

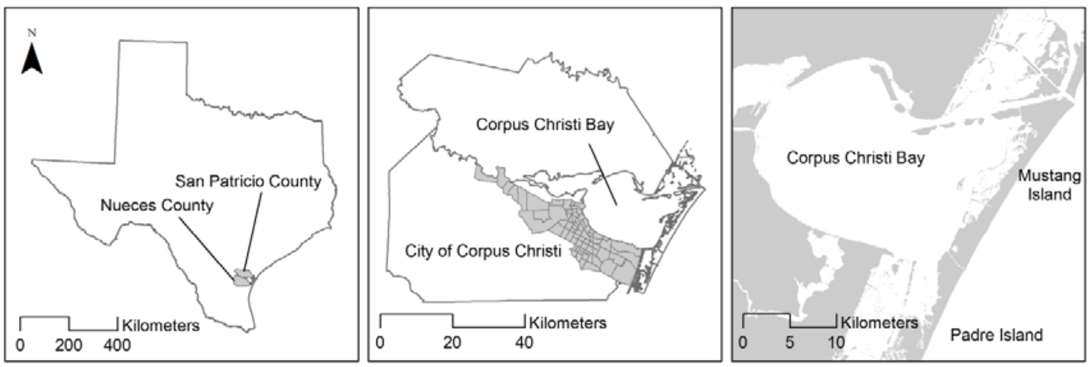


Figure 4-2 Corpus Christi City, Nueces County and San Patricio County

Gulfport City is located on the south coast of Mississippi (Figure 4-3). Harrison County has a population of 194,029 , which includes 70,113 in Gulfport City (U.S. Census Bureau, 2013c).

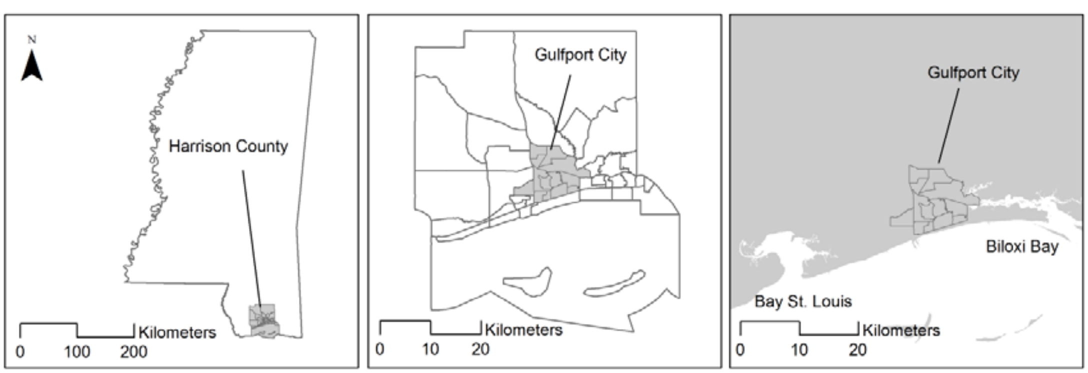


Figure 4-3 Gulfport City and Harrison County

Panama City is located on the north west of Florida (Figure 4-3). Bay County has a population of 174,983, which includes 36,167 in Panama City (U.S. Census Bureau, 2013c).

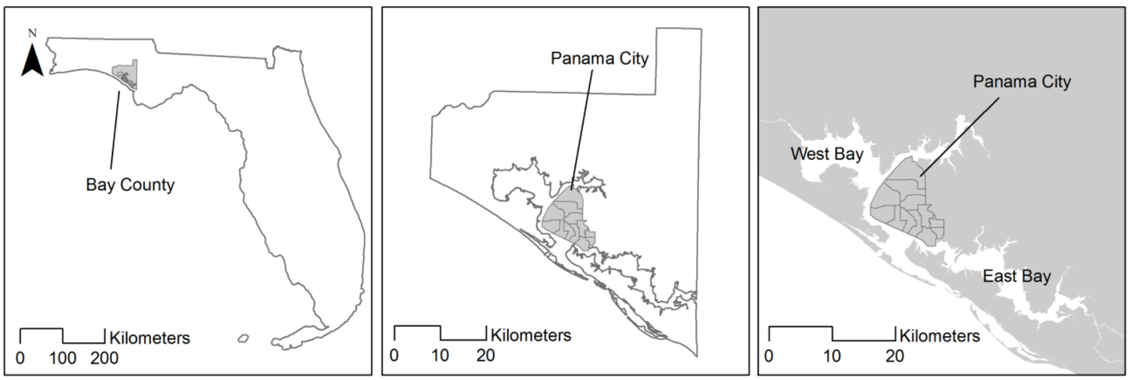


Figure 4-4 Panama City and Bay County

4.2. Parcel Data and Population Data

The US Census Bureau collects and stores census data (e.g. population, ethnics, etc.) at different resolutions, which they call census blocks, census block groups and census tracks. Census blocks have the smallest geographic area, which are usually bound by visible features, such as streets, roads, streams, and railroad tracks, and sometimes nonvisible boundaries. Census blocks are assembled into block groups, which usually contain 600 to 3000 people. Census block groups are assembled into census tracks, which usually contain 1,500 to 8,000 people (U.S. Census Bureau, 2013a). Figure 4-5 through Figure 4-7 shows the population data at census tract, block group and block levels in Corpus Christi city, Gulfport city and Panama city

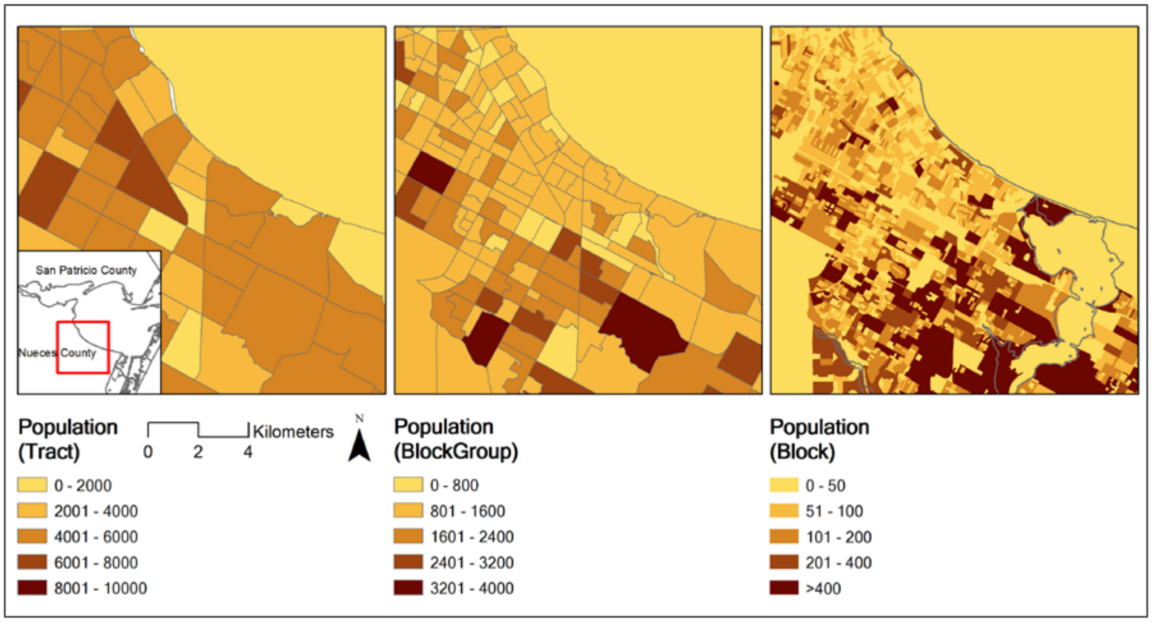


Figure 4-5 Population Data at Census Tract, Block Group, and Block Level (Corpus Christi, TX)

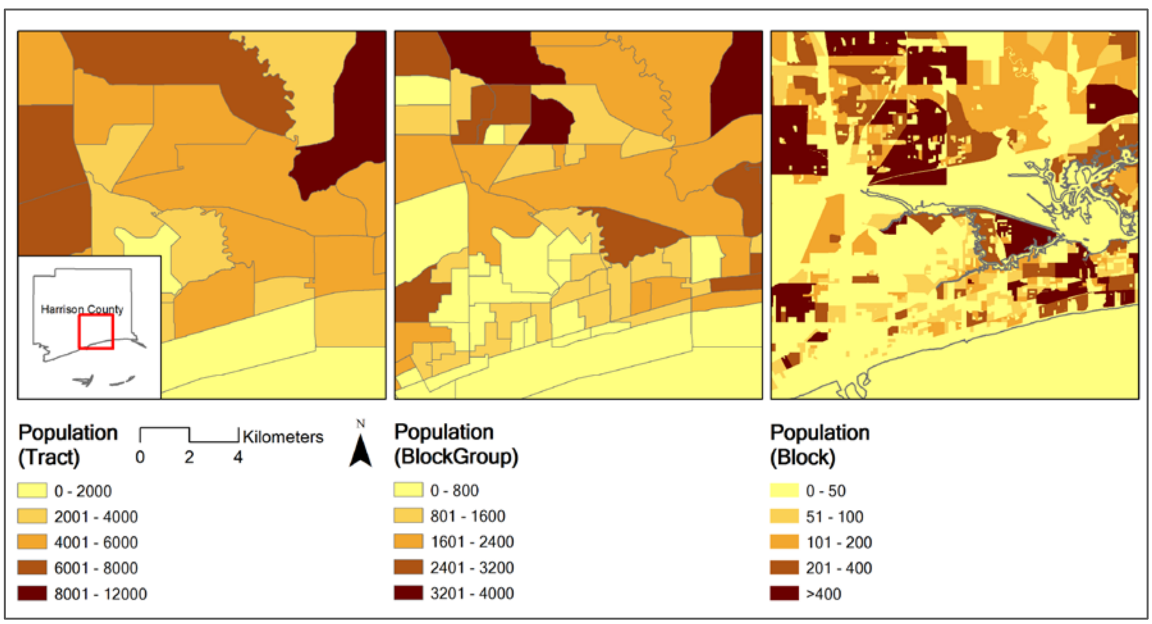


Figure 4-6 Population Data at Census Tract, Block Group, and Block Level (Gulfport, MS)

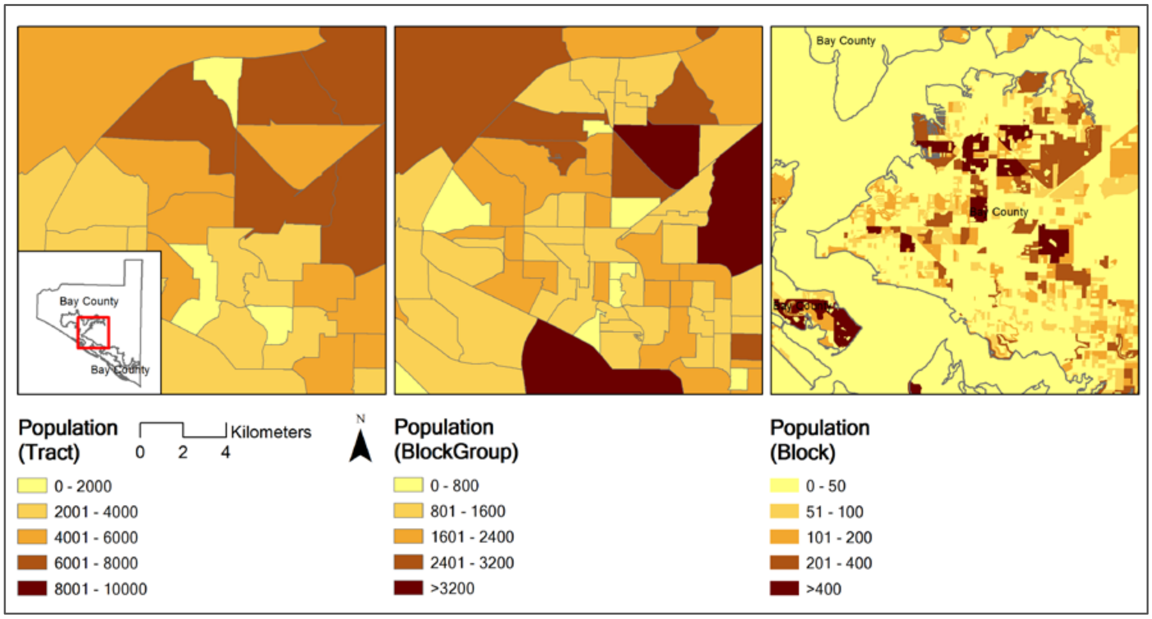


Figure 4-7 Population Data at Census Tract, Block Group, and Block Level (Panama, FL)

Parcel data, which contains the land value, property values, the rights, interests, and ownership of land, is usually managed and maintained by local appraisal districts, is the geographic area defined by the Department of the Interior's Bureau of Land Management (BLM). Governments use parcel data to make decisions about land development, business activities, regulatory compliance, emergency response, and law enforcement. In this study, we obtained parcel data (Figure 4-8) through the County Appraisal District Office of each county or County Official Website (Gulport MS, 2012; Nueces County TX, 2012; Panama FL, 2012; San Patricio County TX, 2012).

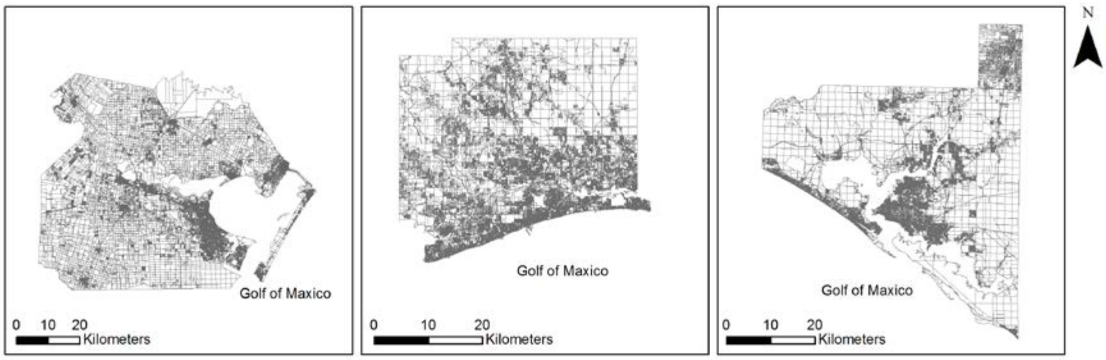


Figure 4-8 Parcels Datasets of Nueces, San Patricio and Harrison Counties

Data examination, by overlaying the satellite imagery with parcel data, is performed to check the potential data problems before conducting damage analysis. It was found that some parcels have houses inside but no value in the attribute table. Although the number of this type of parcels seem not significant compared to the whole dataset (Table 4–1), the location of the parcels are close to the water-prone area (Figure 4-9), which would affect the analysis results. To deal with this problem, the following steps are applied to assign values to the parcels with property value missing:

(1) For every parcel that has value in the raw dataset, calculate the ratio of property value to area.
(2) For each census block, calculate the average value of the "property value –area" ratio.
(3) For a parcel that has property but without property value, identify which census block this parcel belongs to and then multiply the parcel area with the averaged ratio calculated by step (2).

Table 4–1 Parcel Counts for Three Study Areas

	Corpus Christi, TX	Gulfport, MS	Panama, FL
Parcels with properties and values	113,023 (70%)	62,483 (63%)	72,010 (73%)
Parcels that has properties but without values	1,752 (1.08%)	902 (0.91%)	243 (0.25%)
Parcels without properties	46,453 (29%)	35,707 (36%)	25,844 (26.35)
Total	161,228	99,092	98,097

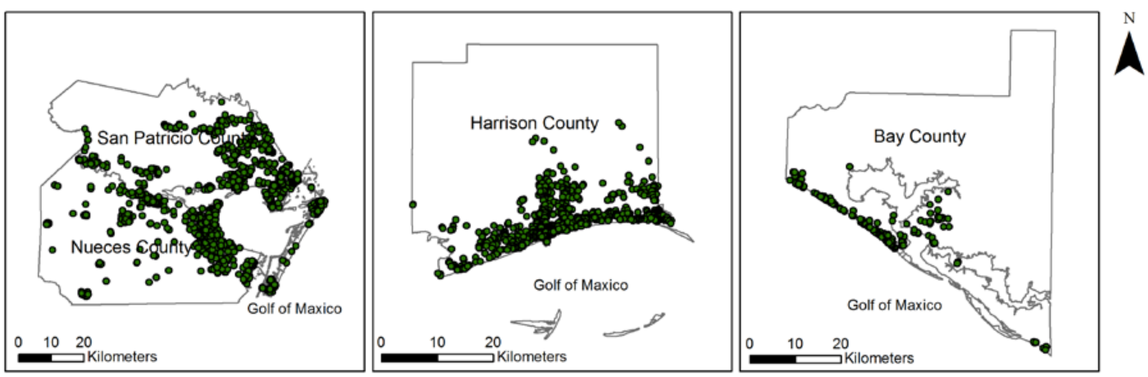


Figure 4-9 Location of Parcels That Have Properties But No Property Value in the Attribute Table

For estimating population being affected by hurricane flooding, we obtain the population data at census block level (U.S. Census Bureau, 2013d), which is the smallest resolution made available by the Census Bureau. Within the census blocks, the population is assumed to be uniformly distributed over the parcels' area, and the population in an individual parcel is indicated in equation (4.1):

$$Population\ in\ parcel = \frac{Area\ of\ parcel}{Total\ area\ of\ parcels\ in\ block} \times Population\ in\ block \qquad (4.1)$$

The spatial analysis was conducted in State Plan projection system for Corpus Christi, Gulfport and Panama City.

4.3. Business Data

The business data is obtained from ReferenceUSA (Infogroup, 2012). The ReferenceUSA database holds business information such as company name, company address, employee size and sales volume, etc. (see Table 3–1 for list of fields). ReferenceUSA business database also has the approximate locations of business, which allows us to georeference them (see Figure 3-9 through Figure 4-12). The NAICS field stands for "North American Industry Classification System", which was developed under the auspices of the Office of Management and Budget (OMB), and adopted in 1997 (U.S. Census Bureau, 2013b). Different number of NAICS digits represents different level of classification of business sectors. Table 4–3 lists 2-digit NAICS code and associated business descriptions[2] (see Appendix D for list of 3-digit NAICS code and business descriptions).

As it takes much computing time and space for showing the results for all businesses, this study selected business data at 3-digit NAICS level, and picked the first four business sectors which are ranked by the employee size (the number of employee), which would account for approximately one-third of the employee at each of the study

[2] The finest level of NAICS contains six digits. For full list of see the NAICS Structure file on U.S. Census.

areas (see table Table 4–4), to conduct the analysis. For business data, we conducted damage analysis using the Return Period Approach and chose employee size, sales volume and affected business as the variables for showing the damage on business due to hurricane surge flood.

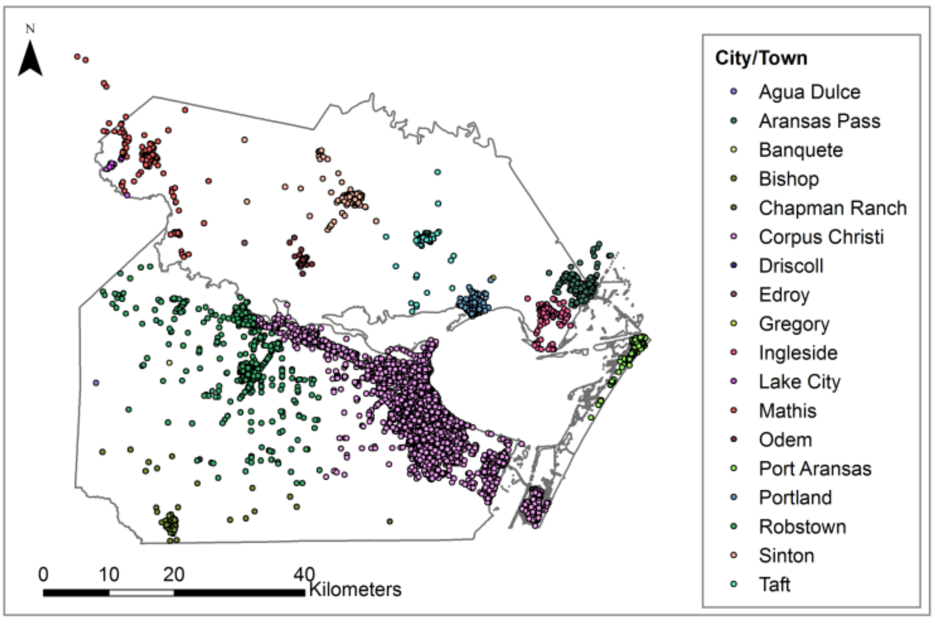


Figure 4-10 Corpus Christi, TX Business Data

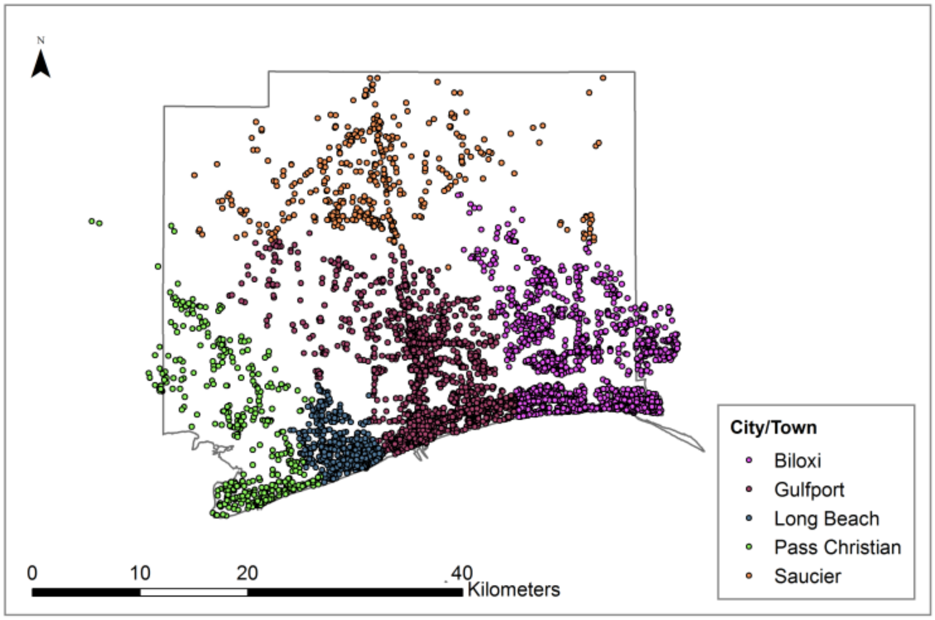


Figure 4-11 Gulfport, MS Business Data

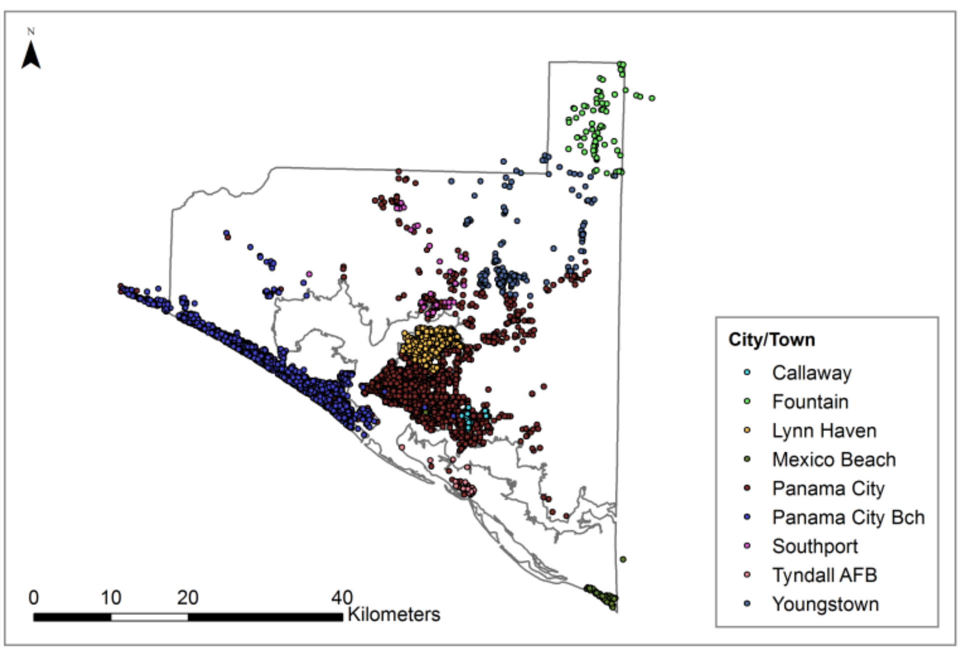


Figure 4-12 Panama, FL Business Data

Table 4–2 Reference USA Business Data Fields

No	**Business Data Field**
1	Company Name
2	Executive First Name
3	Executive Last Name
4	Address
5	City
6	State
7	ZIP Code
8	Credit Score Alpha
9	Executive Gender
10	Executive Title
11	Fax Number Combined
12	IUSA Number
13	Location Employee Size Actual
14	Location Employee Size Range
15	Location Sales Volume Range
16	Location Sales Volume Actual
17	Phone Number Combined
18	Primary SIC Code
19	Primary SIC Description
20	SIC Code 2
21	SIC Code 2 Description
22	Record Type
23	Primary NAICS
24	Primary NAICS Description
25	Latitude
26	Longitude

Table 4–3 Two-Digit NAICS Code and Business Descriptions

NAICS Code	Business description
11	Agriculture, Forestry, Fishing and Hunting
21	Mining, Quarrying, and Oil and Gas Extraction
22	Utilities
23	Construction
31-33	Manufacturing
42	Wholesale Trade
44-45	Retail Trade
48-49	Transportation and Warehousing
51	Information
52	Finance and Insurance
53	Real Estate and Rental and Leasing
54	Professional, Scientific, and Technical Services
55	Management of Companies and Enterprises
56	Administrative and Support and Waste Management and Remediation Services
61	Educational Services
62	Health Care and Social Assistance
71	Arts, Entertainment, and Recreation
72	Accommodation and Food Services
81	Other Services (except Public Administration)
92	Public Administration

Table 4–4 Statistics for Selected Business Data

NAICS (3-digit)	Description	Employee Size	Employee Size (% of total)	Sales Volume (million)	Sales Volume (% of total)	Number of Business	Number of Business (% of total)
Corpus Christi, TX							
722	Food Services and Drinking Places	14,888	8.37%	649	2.13%	984	5.25%
611	Educational Services	14,842	8.34%	25	0.08%	415	2.21%
621	Ambulatory Health Care Services	12,898	7.25%	2,191	7.20%	2,240	11.95%
238	Specialty Trade Contractors	8,968	5.04%	1,237	4.07%	849	4.53%
Total	All businesses	177,867		30,408		18,750	
Gulfport, MS							
713	Amusement, Gambling, and Recreation Industries	13,019	12.51%	879	5.16%	115	0.96%
722	Food Services and Drinking Places	10,369	9.96%	540	3.17%	506	4.24%
621	Ambulatory Health Care Services	6,866	6.60%	1,412	8.28%	1,431	11.99%
611	Educational Services	5,524	5.31%	82	0.05%	177	1.48%
Total	All businesses	104,065		17,042		11,938	
Panama, FL							
722	Food Services and Drinking Places	11,287	11.27%	519	3.09%	690	7.39%
621	Ambulatory Health Care Services	8,870	8.86%	1,709	10.16%	1,094	10.07%
611	Educational Services	5,274	5.27%	29	0.18%	176	1.59%
541	Professional, Scientific, and Technical Services	4,216	4.21%	695	4.13%	1,006	8.07%
Total	All businesses	100,152		16,822		11,917	

4.4. Digital Elevation Model (DEM)

The DEM data is obtained from USGS National Map Viewer website (USGS, 2013). The digital elevation model (DEM) used in this study has a horizontal resolution of 10 m and its elevations were referred to the North American Vertical Datum 1988 (NAVD88) (USGS, 2012). The surge value used for evaluating damage is referred to the mean sea level (MSL). According to NOAA's website-Center for operational oceanographic products and service (NOAA, 2012), the present-day MSL is 0.146m (0.48 feet) higher than NAVD88 at NOAA's station "Corpus Christi, TX" (station number 8775870). The Corpus Christi area DEM used in this analysis was uplifted by 0.146m (0.48 feet). Respectively, the Gulfport area DEM was shifted vertically by 0.172 m based on the NOAA's station "Bay Waveland Yacht Club, MS" (station number 8747437).

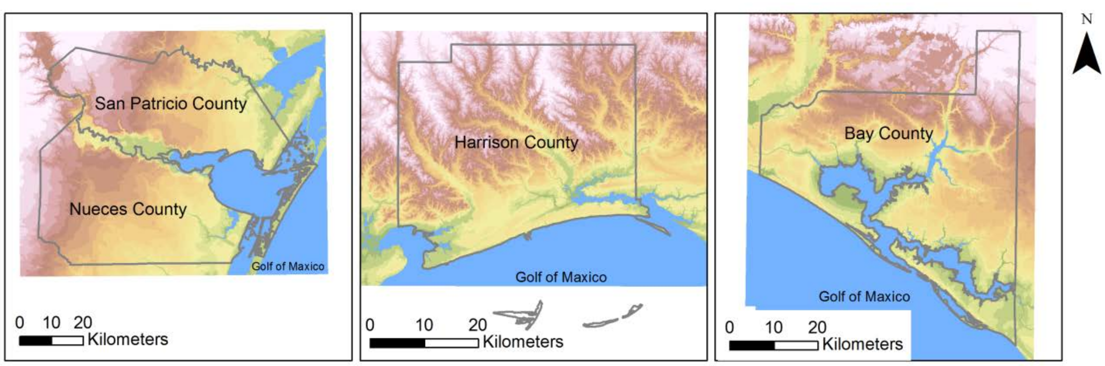


Figure 4-13 Digital Elevation Model of Three Study Area

4.5. Building Count Information at Census Block Level

The building count information is obtained from HAZUS-MH. HAZUS-MH contains census block shape files and the building count table. The raw data of block shape file and building count table are separated but both has census block ID field. By joining the census block ID field, the building count database can be added to the attribute table of the census block shape files. The building count includes 33 types of occupancy, which can be categorized as 7 major types, such as RES (short for residential), COM (short for commercial), etc. (see Table 4–5). Figure 4-14 also shows the ratio of occupancy types for different census blocks.

Table 4–5 Major Types of Occupancy in Hazus Building Count Database

Major Occupancy Type	Description/Purpose	Minor Occupancy Type
RES	Residential	RES1, RES2, RES3A, RES3B, RES3C, RES3D, RES3E, RES3F, RES4, RES5, RES6
COM	Commercial	COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM8, COM9, COM10
IND	Industrial	IND1, IND2, IND3, IND4, IND5, IND6
AGR	Agricultural	AGR1
REL	Religious	REL1
GOV	Governmental	GOV1, GOV2
EDU	Educational	EDU1, EDU2

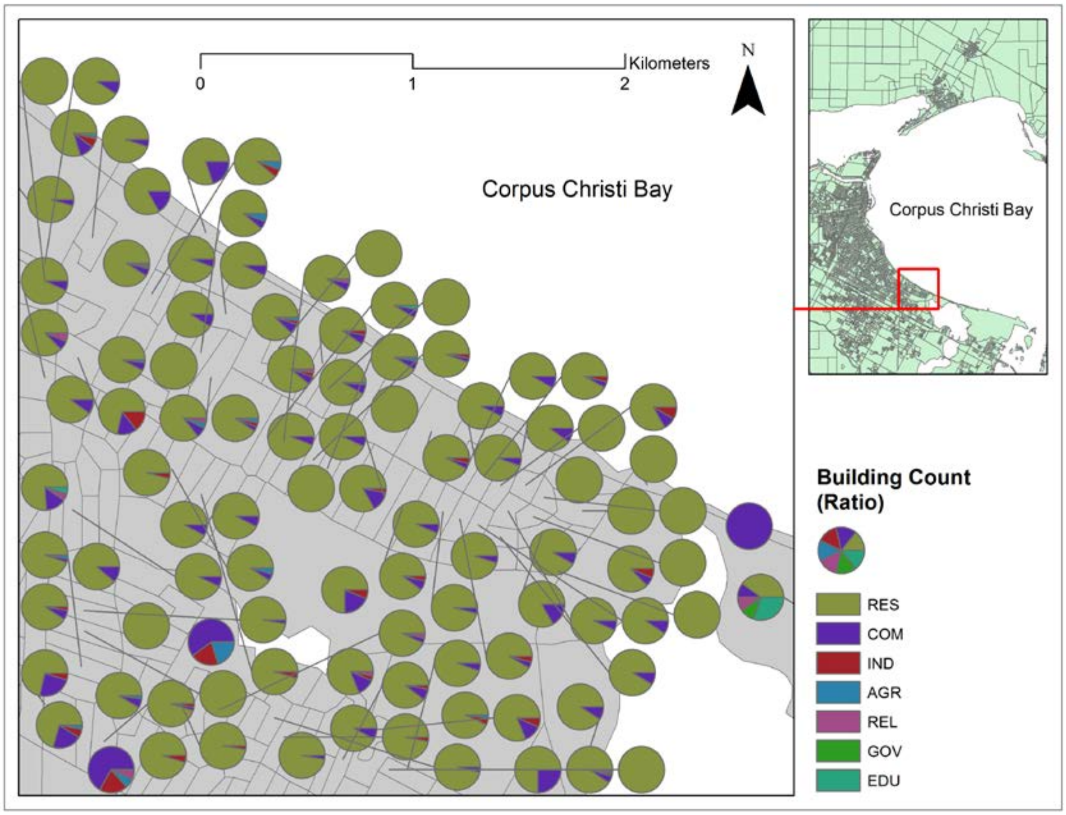


Figure 4-14 HAZUS-MH Census Block Level Building Count Information

4.6. Surge Values at SRF Stations

Surge values were obtained with surge response functions provided by the author of (Irish et al., 2011a). As mentioned above, the locations for which surge responses were provided are called SRF stations (Figure 4-15). For each SRF station, the surge response is stored as a column vector. Each value in the column vector is a function of hurricane parameters (i.e. central pressure, radius, approaching velocity, approaching angle, and landfall location.). Combing the surge matrix for all the SRF stations side by side and put the associated hurricane parameters along with the surge data, the dataset will be like what is shown in Table 4–7. In this format, the hurricane parameters are discretely distributed with specified range and step size.

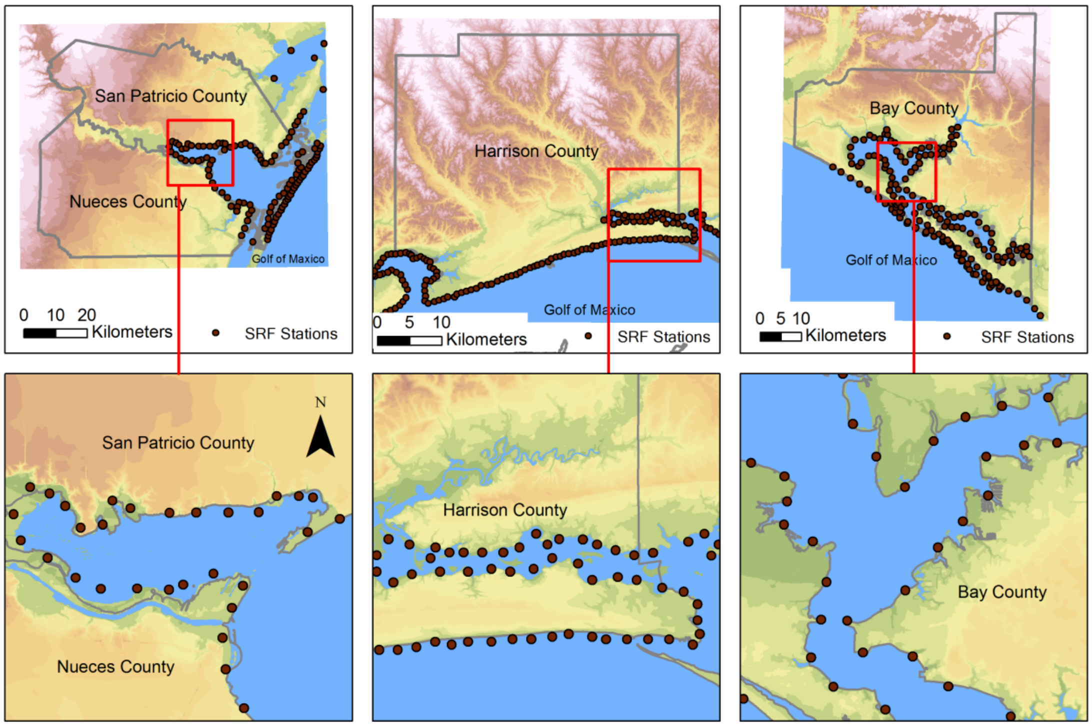


Figure 4-15 SRF Stations for Corpus Christi, Gulfport and Panama City

Table 4–6 Example of Surge Response Data at SRF Stations

C_p	R_p	V_f	θ	x_0	Surge(m) at SRF station 61	Surge(m) at SRF station 62	Surge(m) at SRF station 63	Surge(m) at SRF station 64
770	4	3	-80	-80	1.4931	1.4312	1.7854	1.7345
770	4	3	-80	-75	1.5196	1.457	1.8181	1.7664
770	4	3	-80	-70	1.5471	1.4838	1.8519	1.7995
770	4	3	-80	-65	1.5756	1.5115	1.887	1.8339
770	4	3	-80	-60	1.6051	1.5403	1.9234	1.8696
770	4	3	-80	-55	1.6358	1.5702	1.9613	1.9068
770	4	3	-80	-50	1.6676	1.6013	2.0007	1.9454
(continue)	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯

Note. c_p: Hurricane central pressure; R_p: Hurricane radius to the maximum wind; v_f: Hurricane forward speed; θ: Hurricane approach angle with respect to the shoreline.

Although Resio et al. (2009) identifies five hurricane parameters that influence surge generation, studies that apply surge response functions in conjunction with the joint probability method (Irish et al., 2009; Resio et al., 2009; Irish et al., 2011a) use only the most influential parameters; central pressure, storm size, and landfall location. According to Irish et al. (2008), the effects of forward speed and track angle on surge generation are less influential. In the SRFs of Corpus Christi, all five parameters are used but in Gulfport, only central pressure, radius and landfall location are considered. Table 4–7 and Table 4–8 list the range and step size of the storm parameters for Corpus Christi and Gulfport. The landfall location parameter is denoted as distance from a reference point (at 0 km). Figure 4-16 and Figure 4-17 show the landfall locations for Corpus Christi and Gulfport respectively.

Table 4–7 Hurricane Parameter Range and Step Size for Corpus Christi, TX

Parameters	Range	Step Size
Central Pressure	770 ~ 970 mb	20 mb
Storm size	4 ~ 64 nm	4 nm
Landfall location	-80~80 km	5 km
Forward speed	3 ~21 knots	3 knots
Approaching angle	-80 ~50 degree	10 deg

Table 4–8 Hurricane Parameter Range and Step Size for Gulfport, MS

Parameters	Range	Step Size
Central Pressure	770 ~ 970 mb	20 mb
Storm size	4 ~ 64 nm	4 nm
Landfall location	0~275 km	5 km

Table 4–9 Hurricane Parameter Range and Step Size for Panama, FL

Parameters	Range	Step Size
Central Pressure	770 ~ 970 mb	20 mb
Storm size	4 ~ 64 nm	4 nm
Approaching angle	-80 ~50 degree	10 deg
Landfall location	400~665 km	5 km

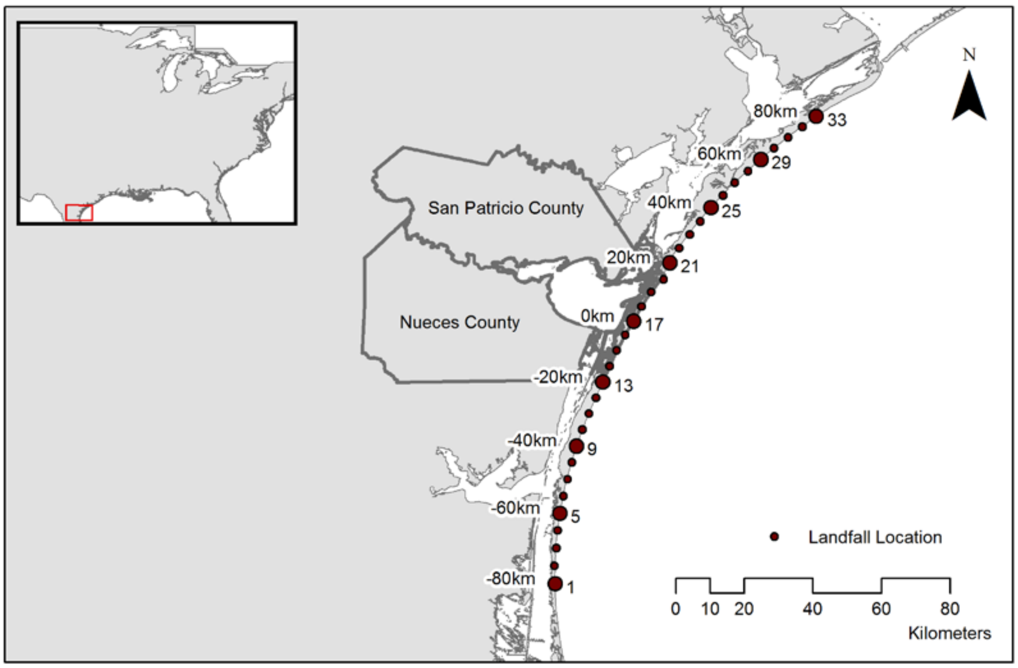


Figure 4-16 Corpus Christi, TX Landfall Locations

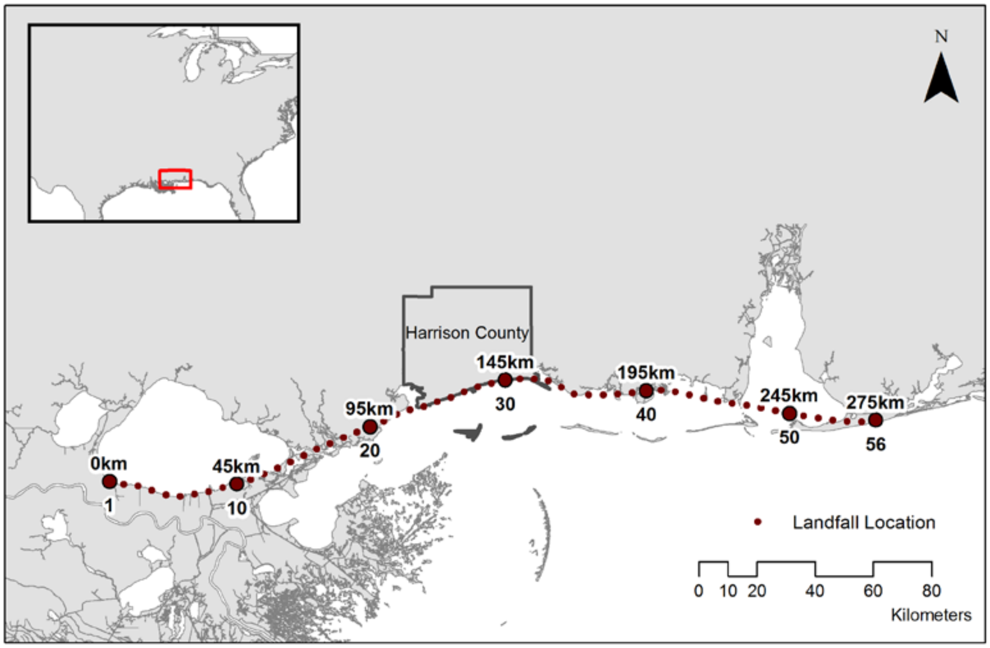


Figure 4-17 Gulfport, MS Landfall Locations

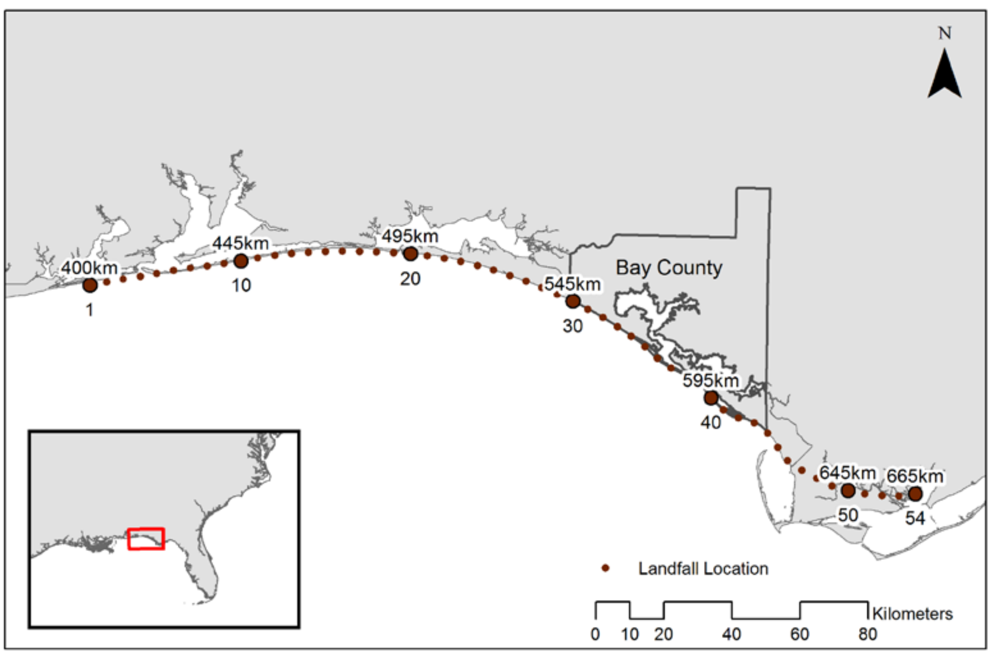


Figure 4-18 Panama, FL Landfall Locations

5. RESULTS

As is discussed in Methodology section, the change of sea surface temperature (SST) could change the hurricane intensity (central pressure). According to the hurricane parameters probability distribution (Eq. 2.3) the overall probability distribution will change. Since the surge value is a function of hurricane parameters, the surge probability will also change. This section discusses the impact of SST on surge probability distribution change in terms of expected surge value shift. The change of surge probability distribution will also result in different amount of damage, which will be discussed in 5.3 and 5.4.

5.1. The Effect of SST on Surge Values

For the selected IPCC scenarios along with specified SLR and SST change, the surge probability density functions for the a SRF station in years 2030 and 2080 are similar to what Figure 5-1 shows. (For the surge probability density functions of all SRF stations, see Appendix F.)

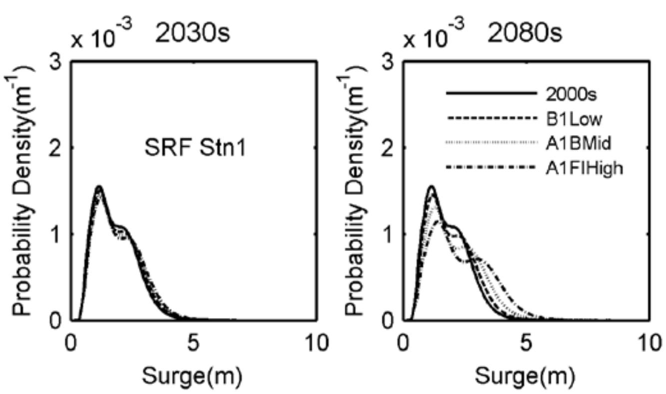


Figure 5-1 Surge Probability Distribution

It was found that, due to the effect of SST and the resulting change in hurricane central pressure probability, the surge probability distribution tends to shift to higher surge values and the peak of the distribution to drop. To investigate how the surge probability distributions are different for all SRF stations, the expected surge values are shown in Figure 5-2 through Figure 5-4. Note that the expected surges here do not account for the probability of hurricane events that make landfall outside the specified landfall range (Figure 4-13 through Figure 4-15). In other words, this is the expected surge value given that a hurricane makes landfall within the specified landfall range.

Figure 5-2 through Figure 5-4 show that, under current climate conditions (0 m SLR and 0 ℃ ΔSST), the expected surge value at Corpus Christi ranges from 0.5 m to 2.8 m with average about 2 m, while the expected surge value ranges from 1.2 m to 3 m with average about 2.2 m for Gulfport, and ranges from 0.6 to 2.8 with average about 2 m for Panama. Figure 5-5 through Figure 5-7 show that, with the effect of increased SST, the expected surge value will increase by up to 0.2 m in 2030 and up to 0.6 m in 2080 at Corpus Christi; increase by up to 0.22 m in 2030 and up to 0.6 m in 2080 at Gulfport; and increase by up to 0.5 in 2030 and up to 1.5 in 2080 at Panama. Overall, as the SLR and SST increases, the expected surge incensement distribution for all SRF stations tend to spread a wider range, which means the effect on the SRF stations are not the same. In the case of Corpus Christi, it was observed that, the expected surge increase around the barrier island is greater than at other locations, which indicates that the presence of the barrier islands play an important role on surge attenuation in the bay. This phenomenon is not observed in the other two cities where there are no barrier islands.

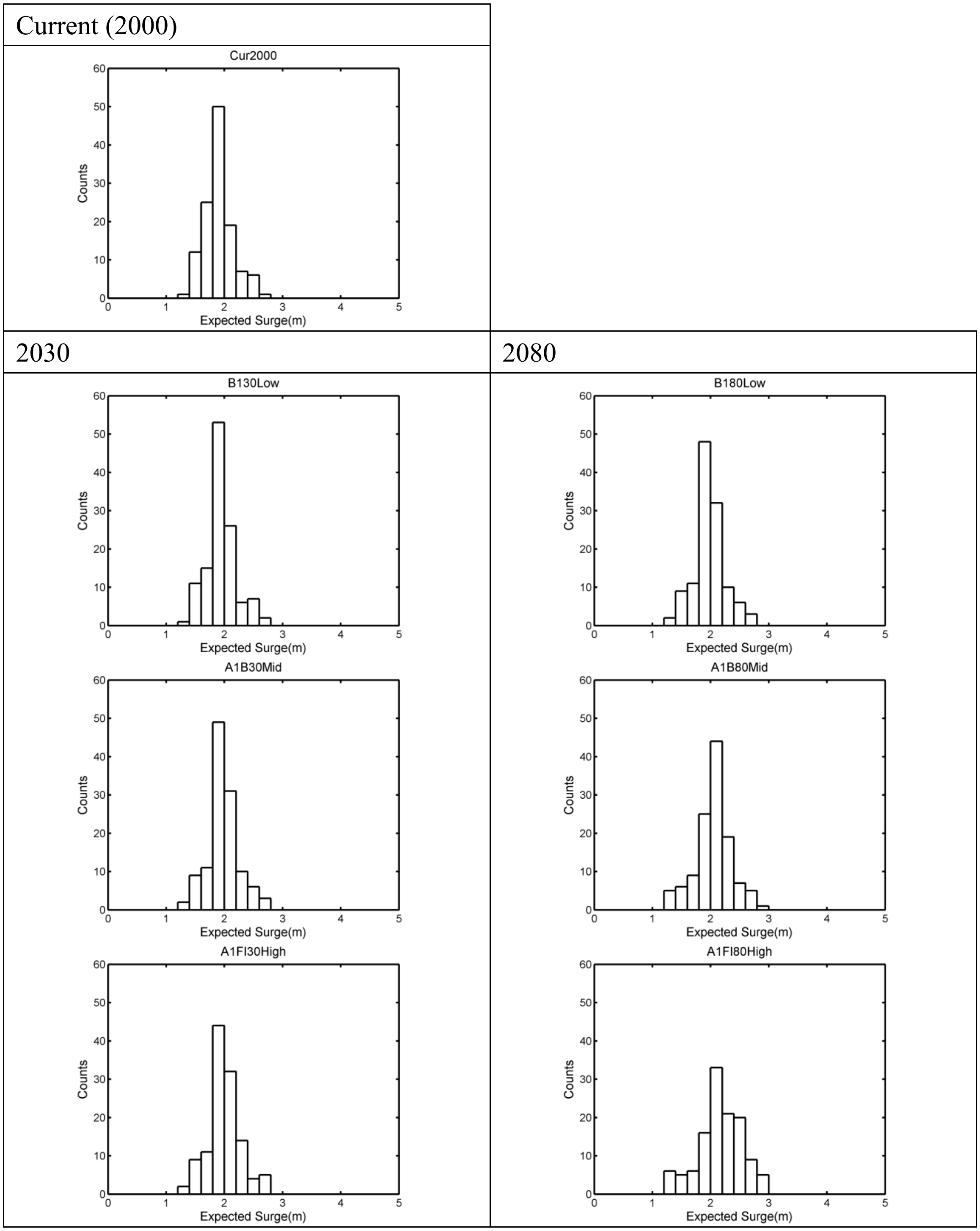


Figure 5-2 Expected Surge Value Distribution (Corpus Christi, TX)

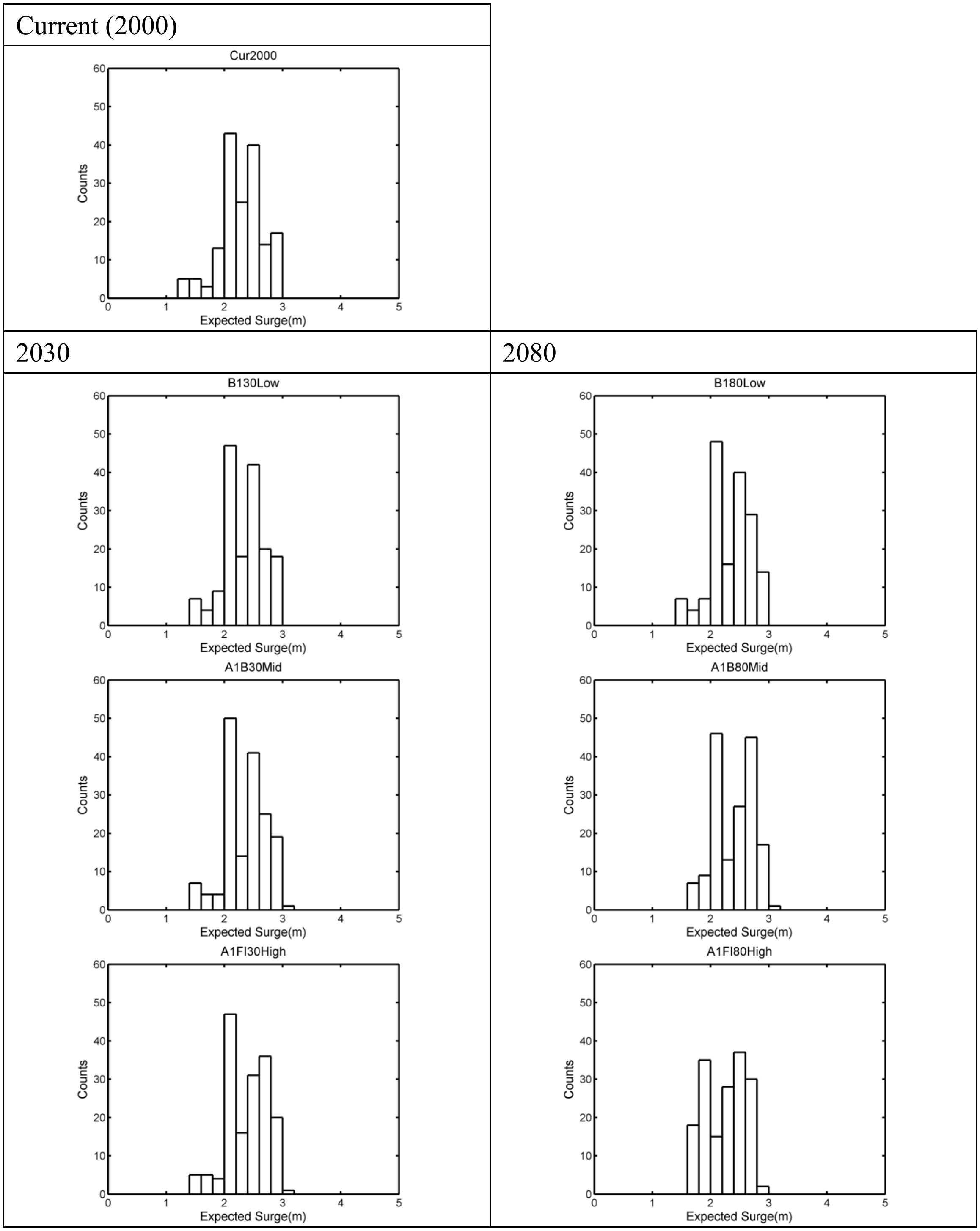


Figure 5-3 Expected Surge Value Distribution (Gulfport, MS)

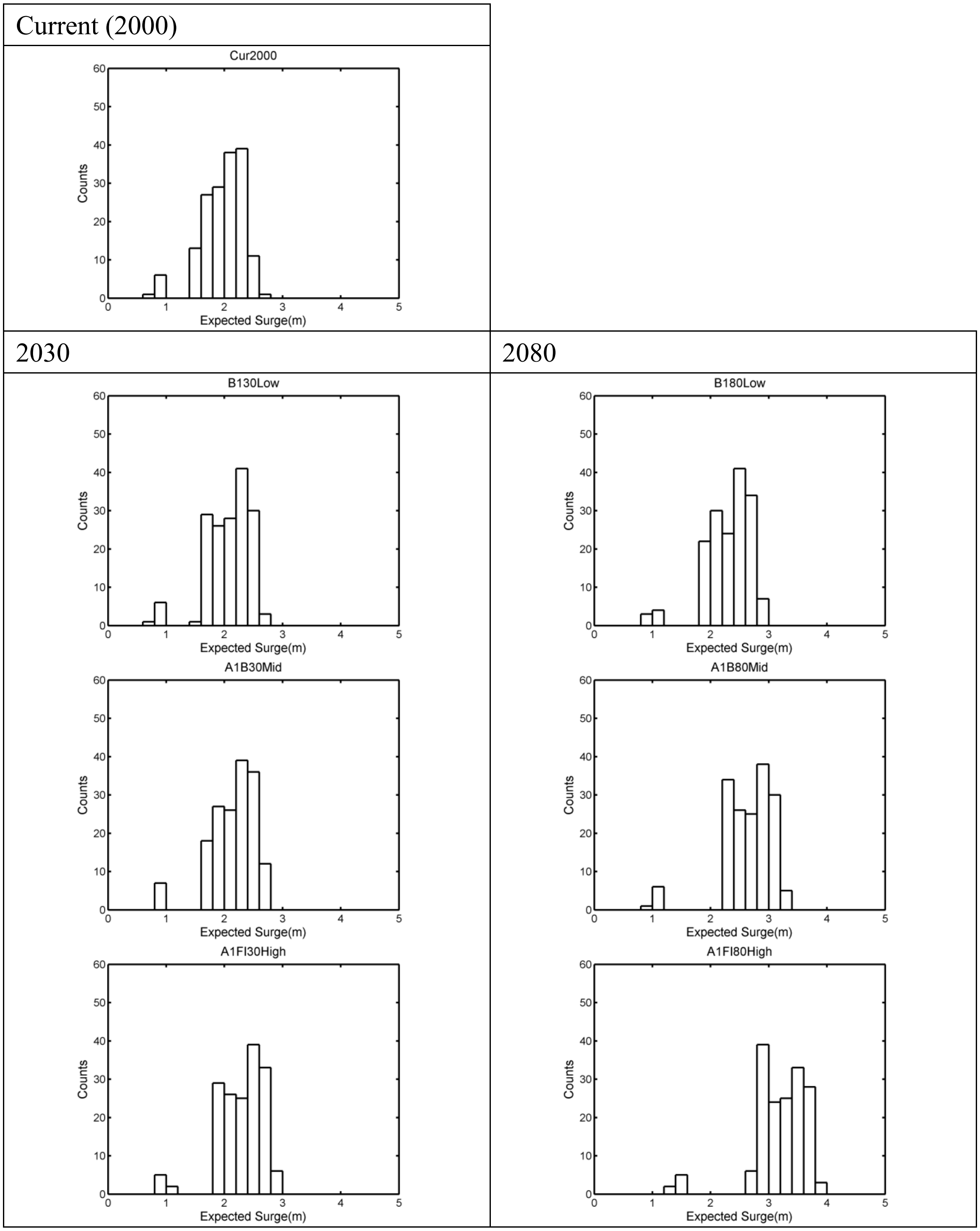


Figure 5-4 Expected Surge Value Distribution (Panama, FL)

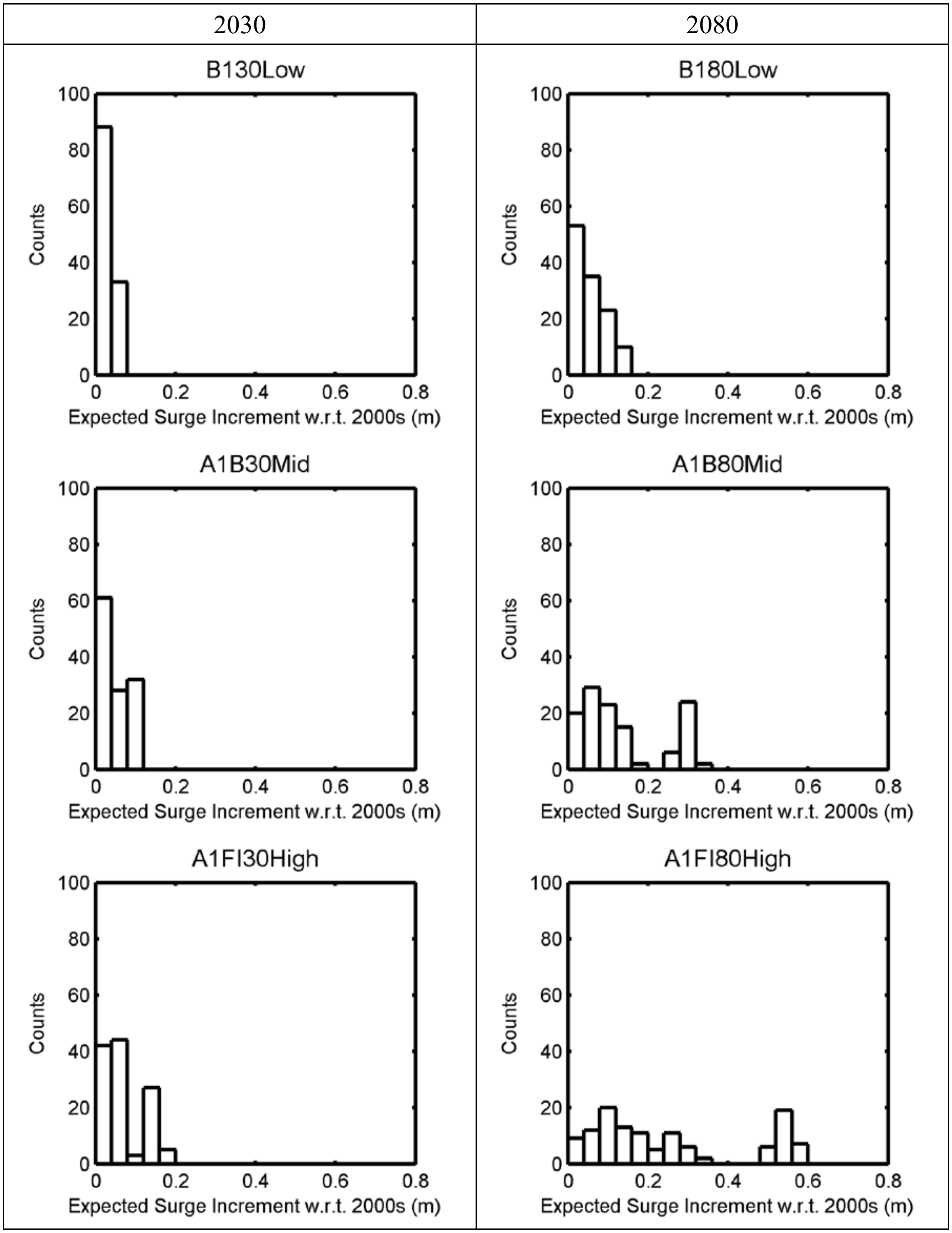


Figure 5-5 Expected Surge Increment Distribution (Corpus Christi City, TX)

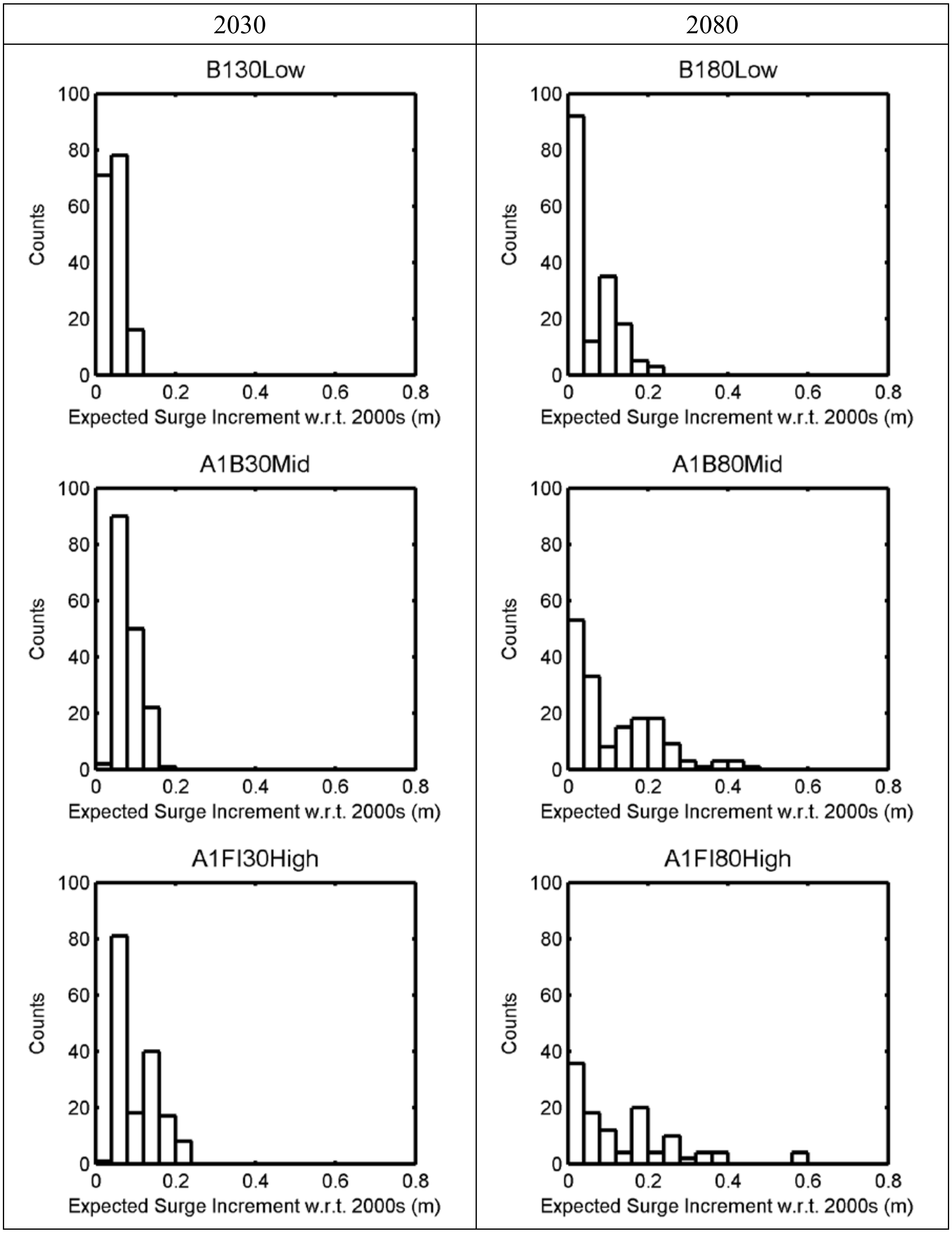


Figure 5-6 Expected Surge Increment Distribution (Gulfport City, MS)

2030	2080

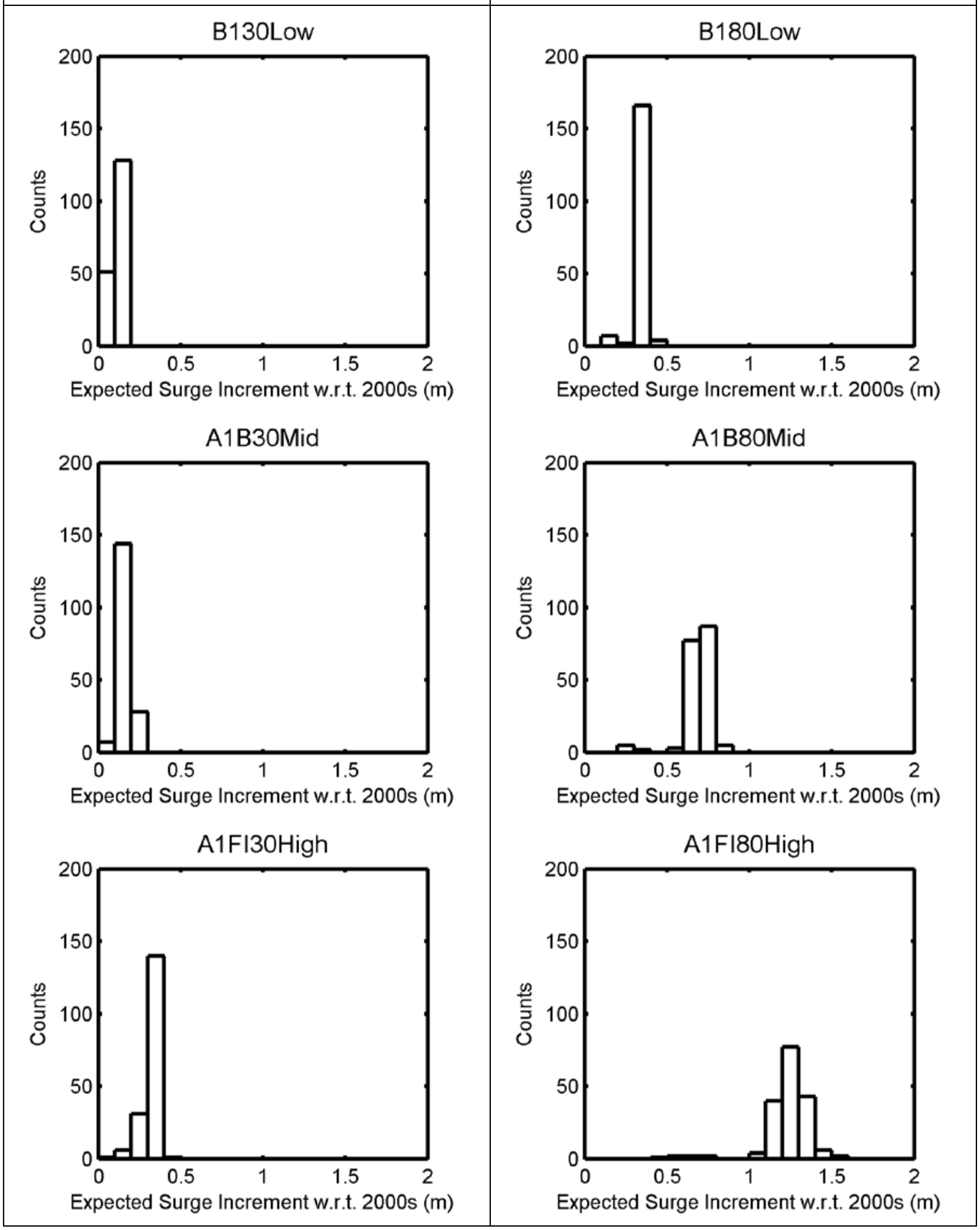


Figure 5-7 Expected Surge Increment Distribution (Panama City, FL)

5.2. The Effect of SLR on Damage Assessment

The analysis presented here assumes no change in the parcel and business data from current time to 2080. Hence, the future sea level will affect the baseline of the damage calculation. In some cases, part of the land and properties will be more adjacent to or even under the projected future sea level (Figure 5-8). In our analysis, the parcels and businesses permanently under sea level will be considered as complete loss and the property value will be consider as the loss value. For example, if a parcel is below the sea level for a certain climate scenario at certain time in the future, the property value of this parcel is considered as the loss, and the population within the parcel is consider as being affected. Figure 5-9 to Figure 5-11 show the total value of properties, population and flooded area of the parcels under sea level for each future climate scenarios we have selected. It was found that the inundated property value of Gulfport is mostly sensitive to SLR comparing to the other two cities.

For the scenarios in year 2030, the inundated property value due to SLR ranges from 8.4 to 13 million at Corpus Christi; ranges from 17 to 18 million dollars Gulfport; ranges from 41 to 42 million dollars at Panama. For the scenarios at 2080, the inundated property value due to SLR ranges from 40 to 170 million dollars at Corpus Christi; ranges from 18 to 420 million at Gulfport; ranges from 42 to 52 million dollars at Panama. The affected population is mostly sensitive to SLR at Corpus Christi, TX. In 2030, the estimation of affected population ranges from 10 to 80 at Corpus Christi; ranges from 90 to 100 at Gulfport; ranges from 210 to 220 at Panama; in 2080, the estimation of affected population ranges from 200 to 1900 at Corpus Christi; from 100 to 770 at Gulfport; from 230 to 800 at Panama. The flooded area is mostly sensitive to SLR at Corpus Christi. In 2030, the estimated flooded area ranges from 13 to 14 km^2 at Corpus Christi; from 4 to 5 km^2 at Gulfport; from 1.5 to 1.6 km^2 at Panama; in 2080, the estimated flooded area ranges from 16 to 38 km^2 at Corpus Christi; from 5 to 17 km^2 at Gulfport; from 3.8 to 21 km^2 at Panama.

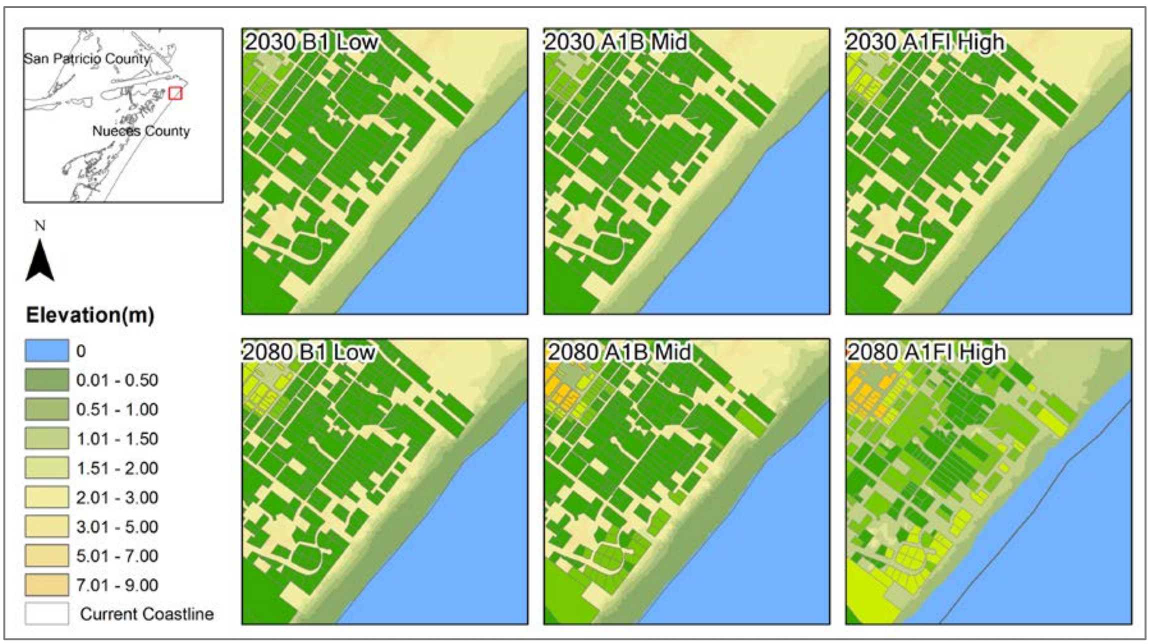


Figure 5-8 DEM for Selected Future Climate Scenarios

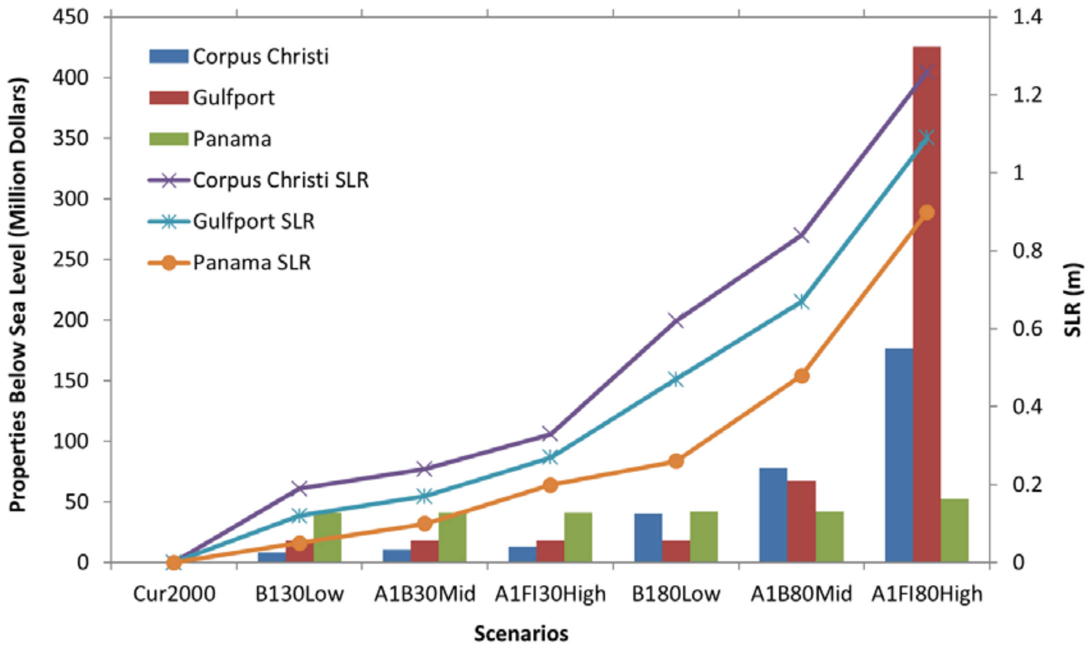


Figure 5-9 Values of Properties below Sea Level for Selected Future Climate Scenarios

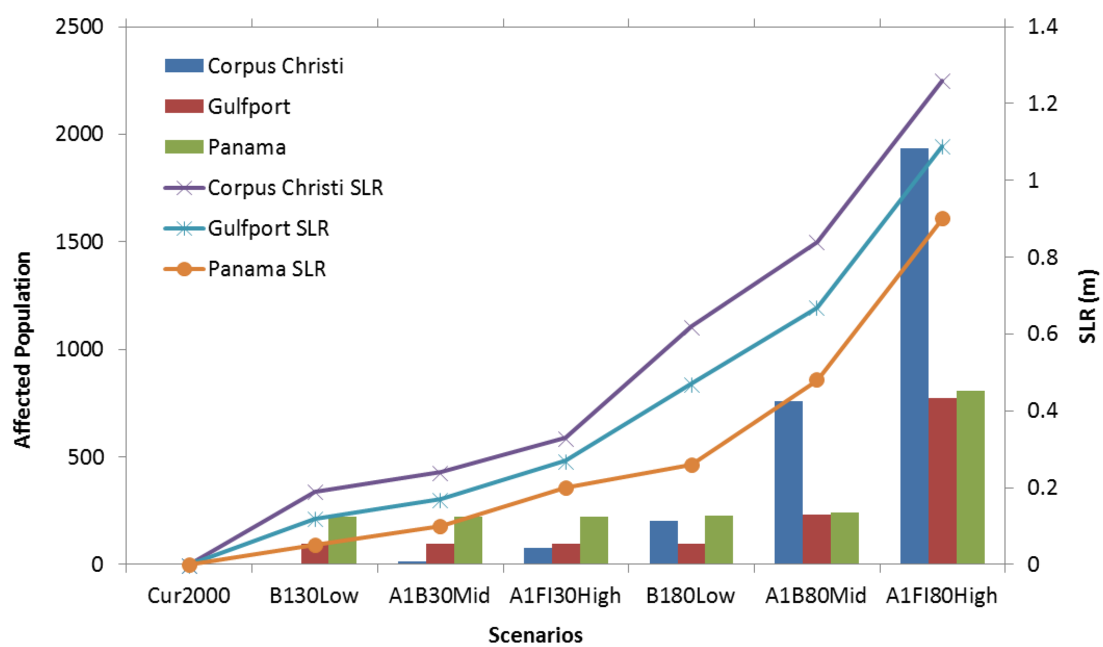


Figure 5-10 Affected Population for Locations below Sea Level for Selected Future Climate Scenarios

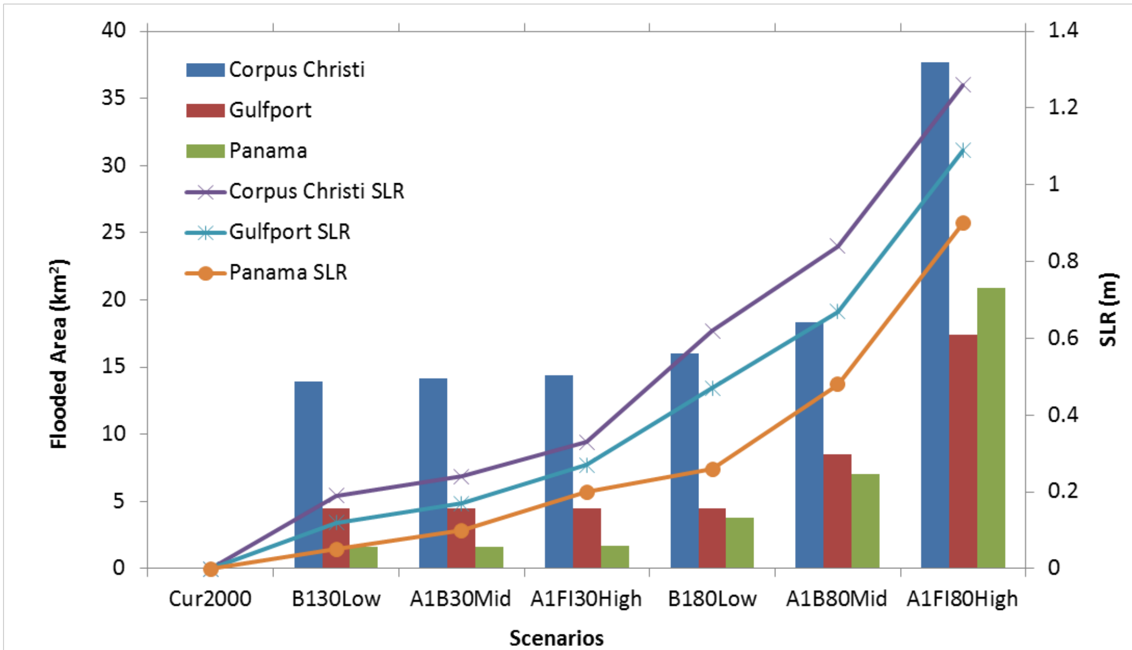


Figure 5-11 Flooded Area below Sea Level for Selected Future Climate Scenarios

5.3. Mapping Expected Annual Loss

Using the approach for calculating the expected annual loss discussed in the Methodology section, the maps of expected annual loss for three study areas are produced (see Figure 5-12 through Figure 5-14). It was found that the most vulnerable area (prone to flood damage) for Corpus Christi (Nueces County and San Patricio County, TX) is the north and south parts of the barrier island; the most vulnerable area for Gulfport (Harrison County, MS) are areas around Bay St. Louis and Biloxi Bay; and the most vulnerable area for Panama City are areas around St. Andrew bay and North Bay (Bay County, FL). Though the result is intuitive as these areas are water prone areas and population/buildings are concentrated in these areas, these maps not only show where the damage would occur but also show how much the expected damage would be.

Figure 5-15 through Figure 5-26 show the expected annual loss value for these vulnerable areas more in detail at both parcel and census block levels. Note that for some future scenarios, some parcels will be under sea level due to the sea level rise, such as the parcels shown in read which locate at the south side of Corpus Christ barrier island, TX (Figure 5-17 and Figure 5-18) and the places around the St. Louis Bay, MS (Figure 5-19 and Figure 5-20). These parcels are considered complete loss, which means the values of these parcels are considered as the lost value. The census block level expected damage is made by summing up the expected parcel damage within the same census block. The results for different future climate scenarios show that the expected annual loss will increase due to sea level rise and the increased surge. These maps not only help us to locate the property at risk but also quantify the damage value the residents would get annually. The comparison of the parcel and block level maps show that while maps at block level give us a sense of which areas are more subject to flood damage; however, it's sometimes misleading because in some blocks there are only a few parcels but the color is full of the entire polygon. For example, there are many empty (white color) areas in Figure 5-19, which cannot be shown in the block level maps (Figure 5-20). On the contrary, the parcel level map cannot provide the expected annual loss values at locations without parcels, while the block level maps can cover a broader area.

The maps at parcel level for current and future climate scenarios show that, under current climate condition, most of the parcels at Corpus Christ barrier island is subject up to 0.6 % of loss, while some parcels at the other two cities could subject up to 2.4 %. The comparison of the maps for the future scenarios shows that the effect of climate change has more impact at Gulfport and Panama than at Corpus Christi. If not consider the parcels under the future sea level (marked in red), the expected annual loss at Corpus Christi increases up to 1.2%, while the value increases up to 3 % at Gulfport and Panama.

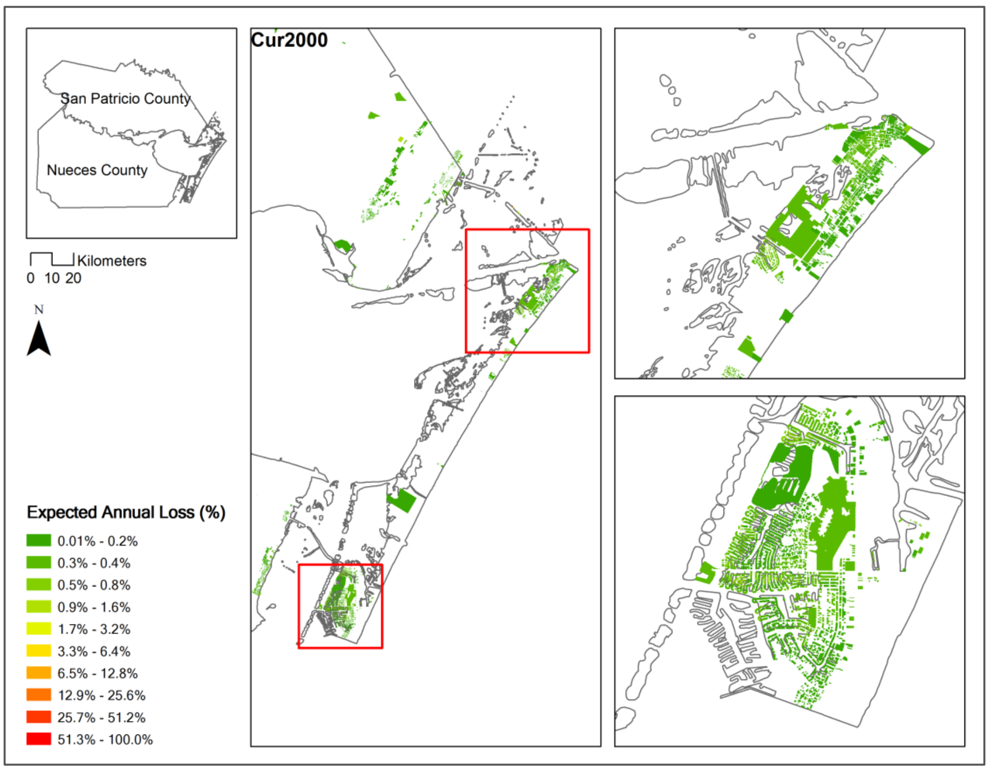


Figure 5-12 Vulnerable Areas of Corpus Christi, TX

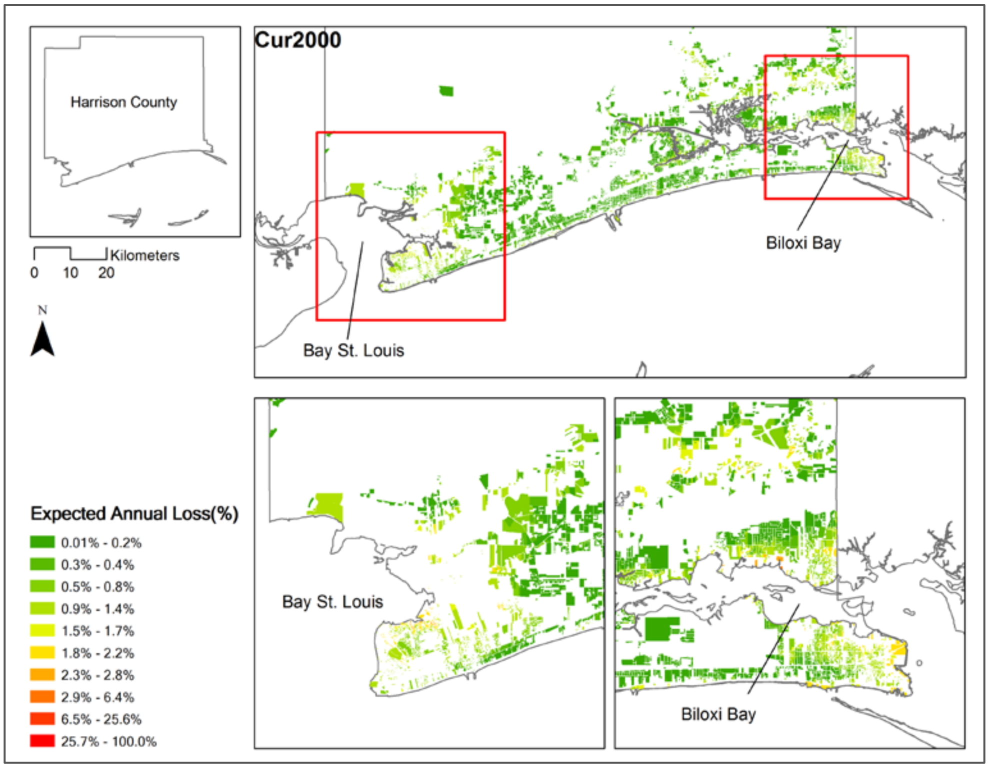


Figure 5-13 Vulnerable Areas of Gulfport, MS

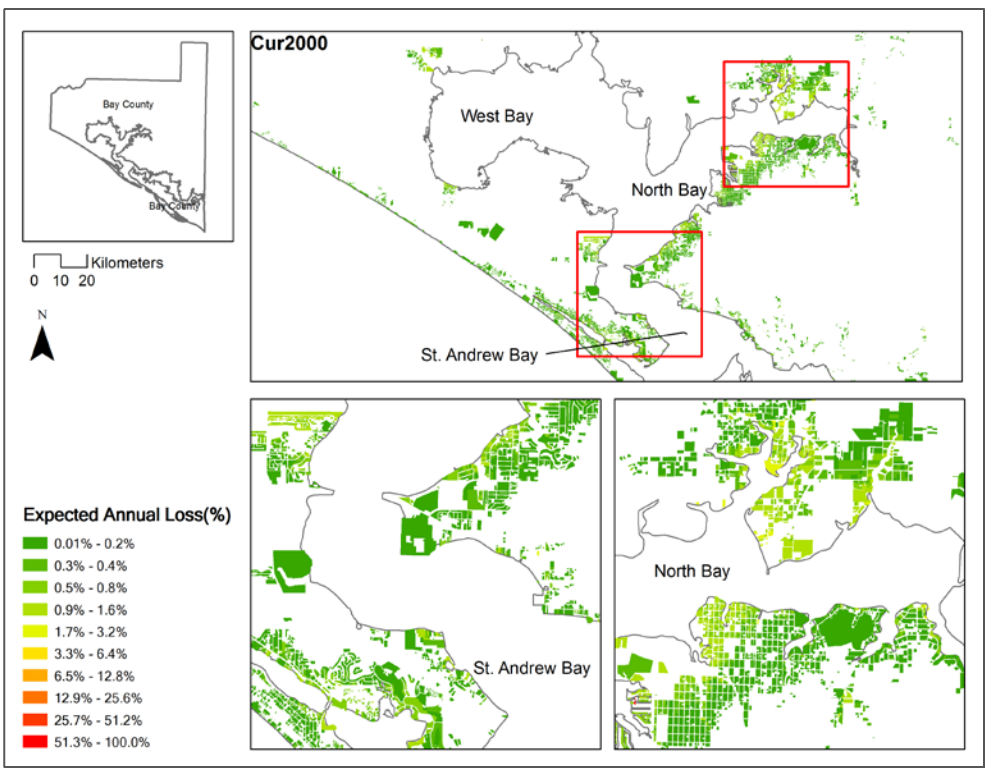


Figure 5-14 Vulnerable Areas of Panama City, FL

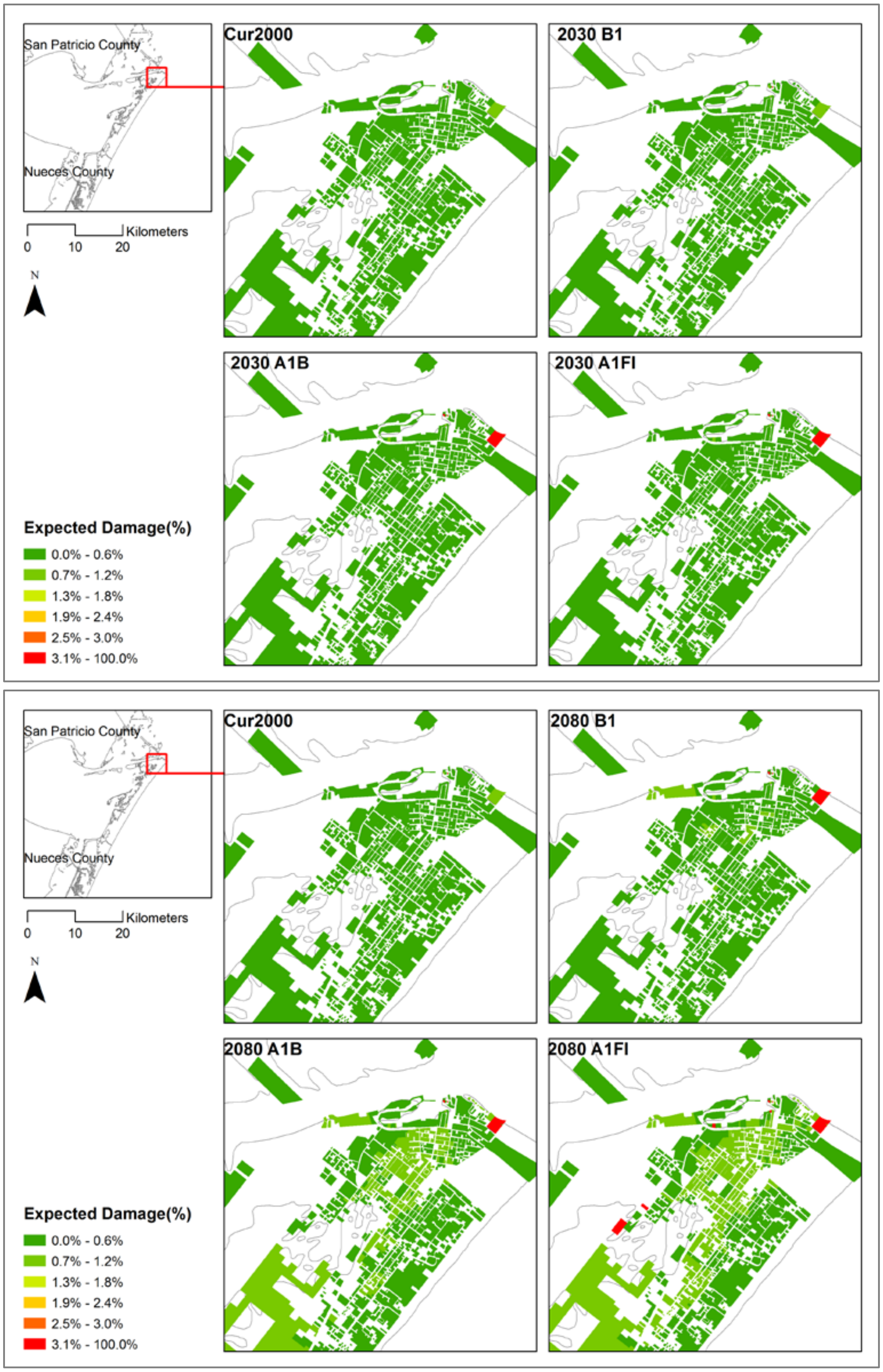


Figure 5-15 Expected Annual Loss at Parcel Level (Port Aransas, TX)

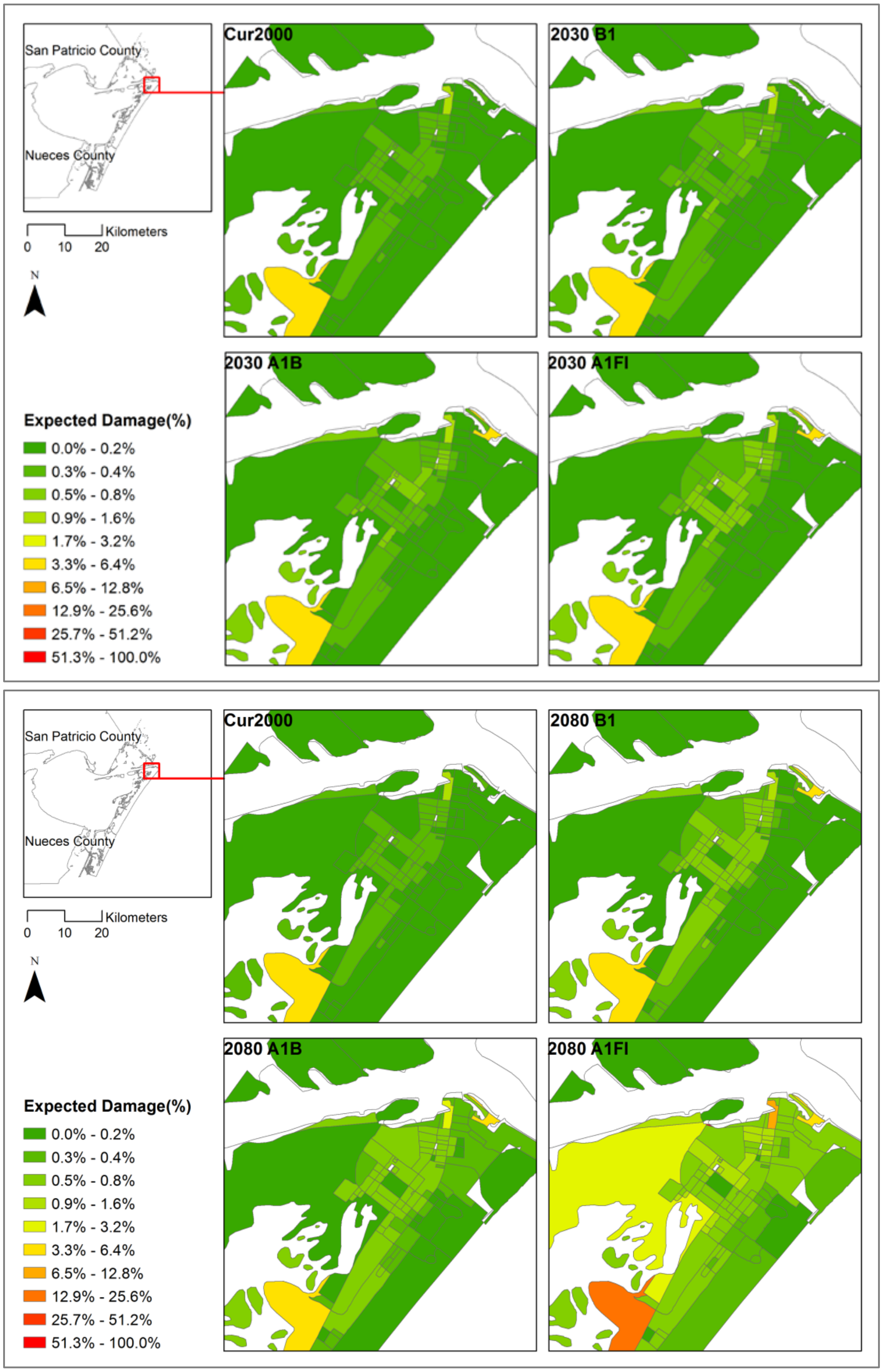


Figure 5-16 Expected Annual Loss at Census Block Level (Port Aransas, TX)

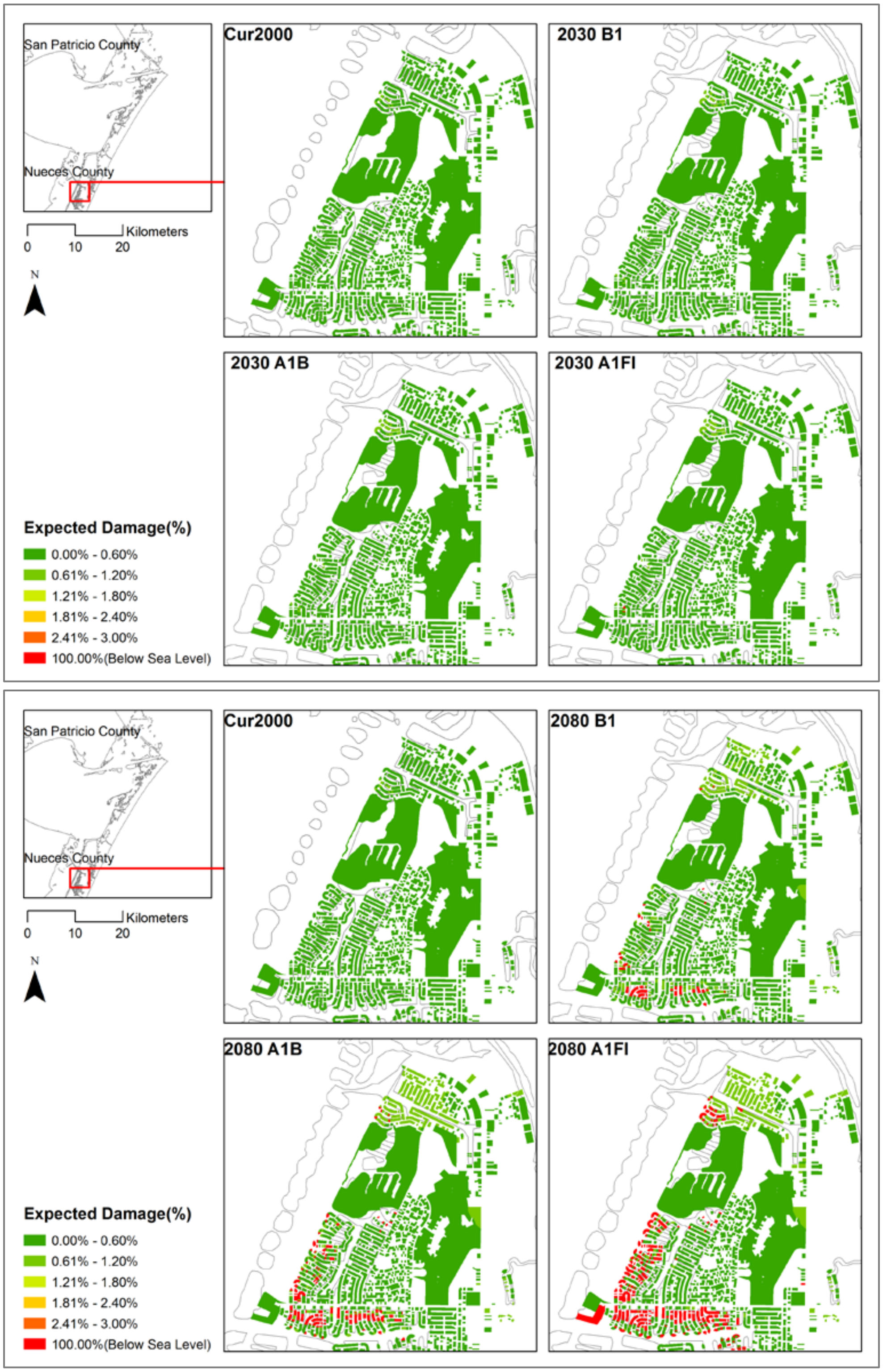


Figure 5-17 Expected Annual Loss at Parcel Level (South Barrier Island, Corpus Christi, TX)

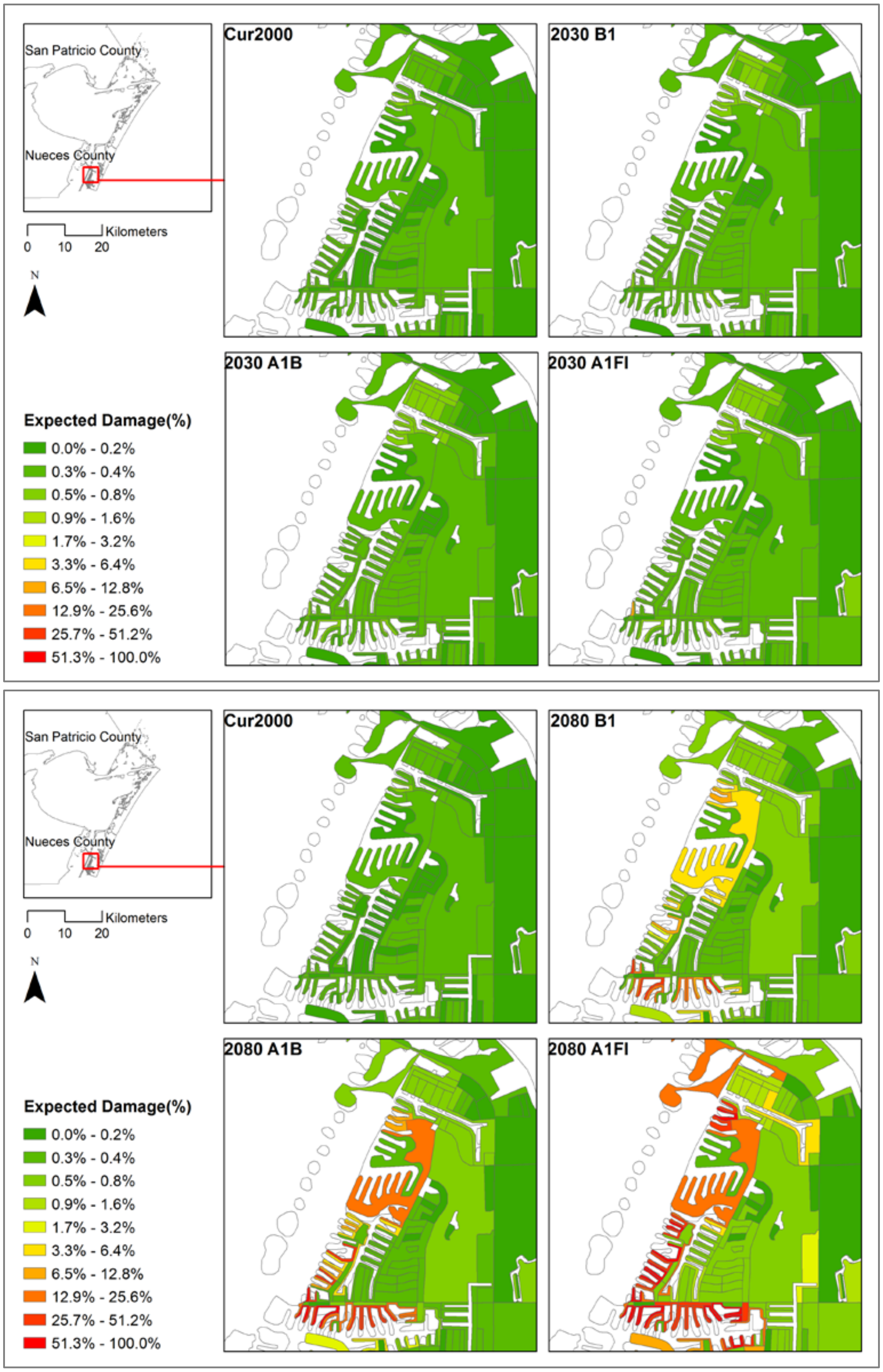


Figure 5-18 Expected Annual Loss at Census Block Level (South Barrier Island, Corpus Christi, TX)

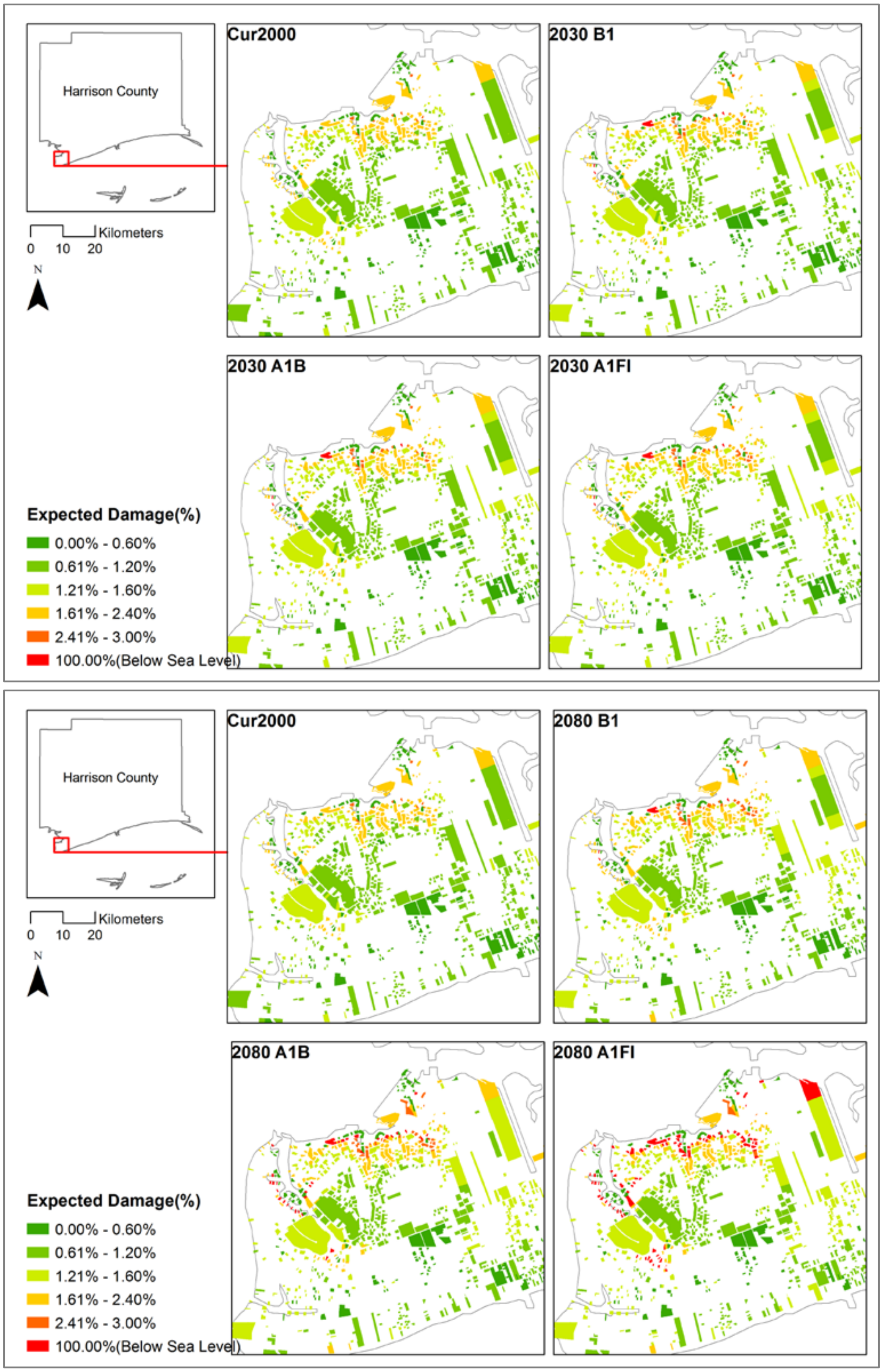


Figure 5-19 Expected Annual Loss at Parcel Level (St. Louis Bay, Gulfport, MS)

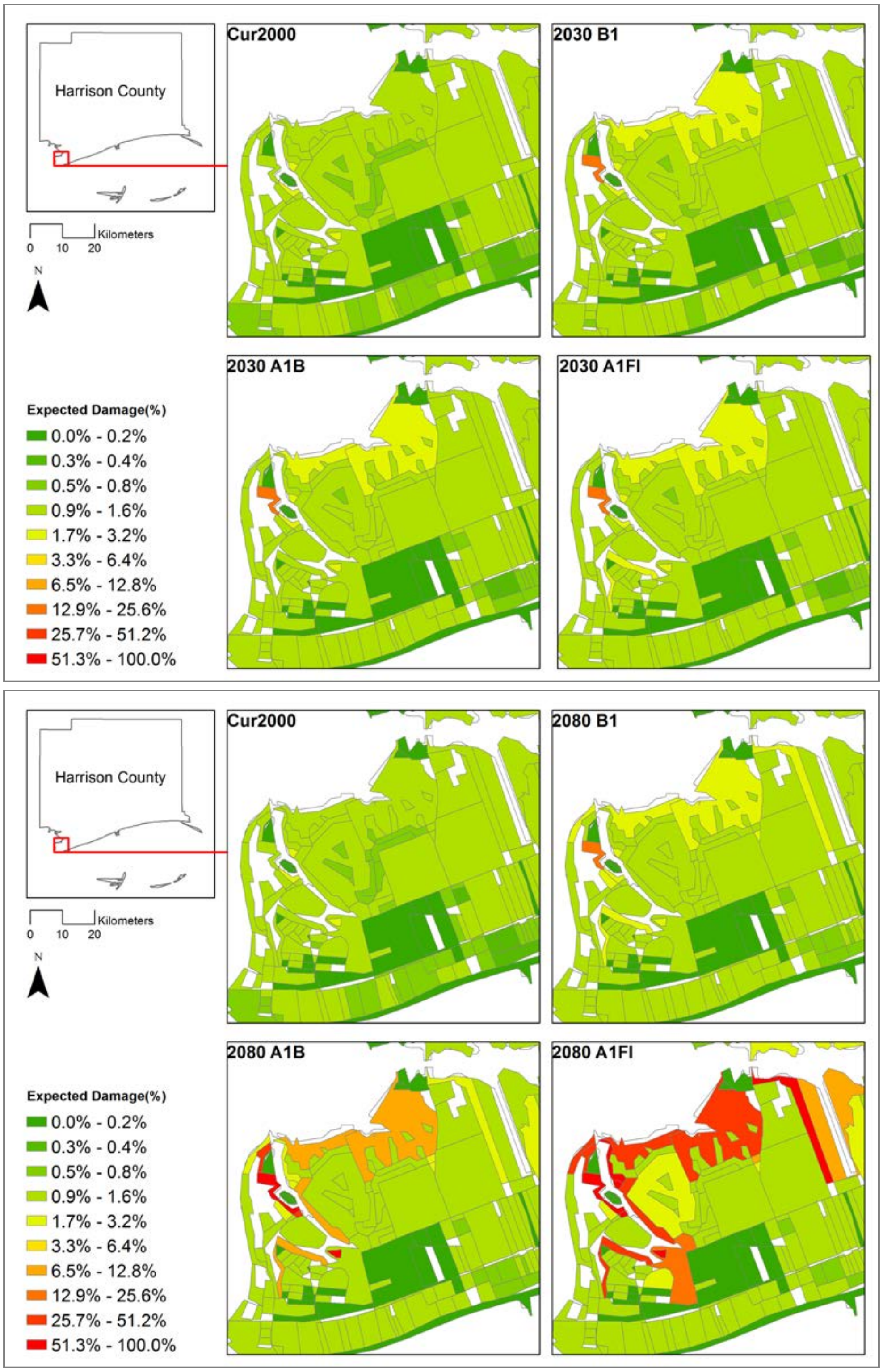


Figure 5-20 Expected Annual Loss at Census Block Level (St. Louis Bay, MS)

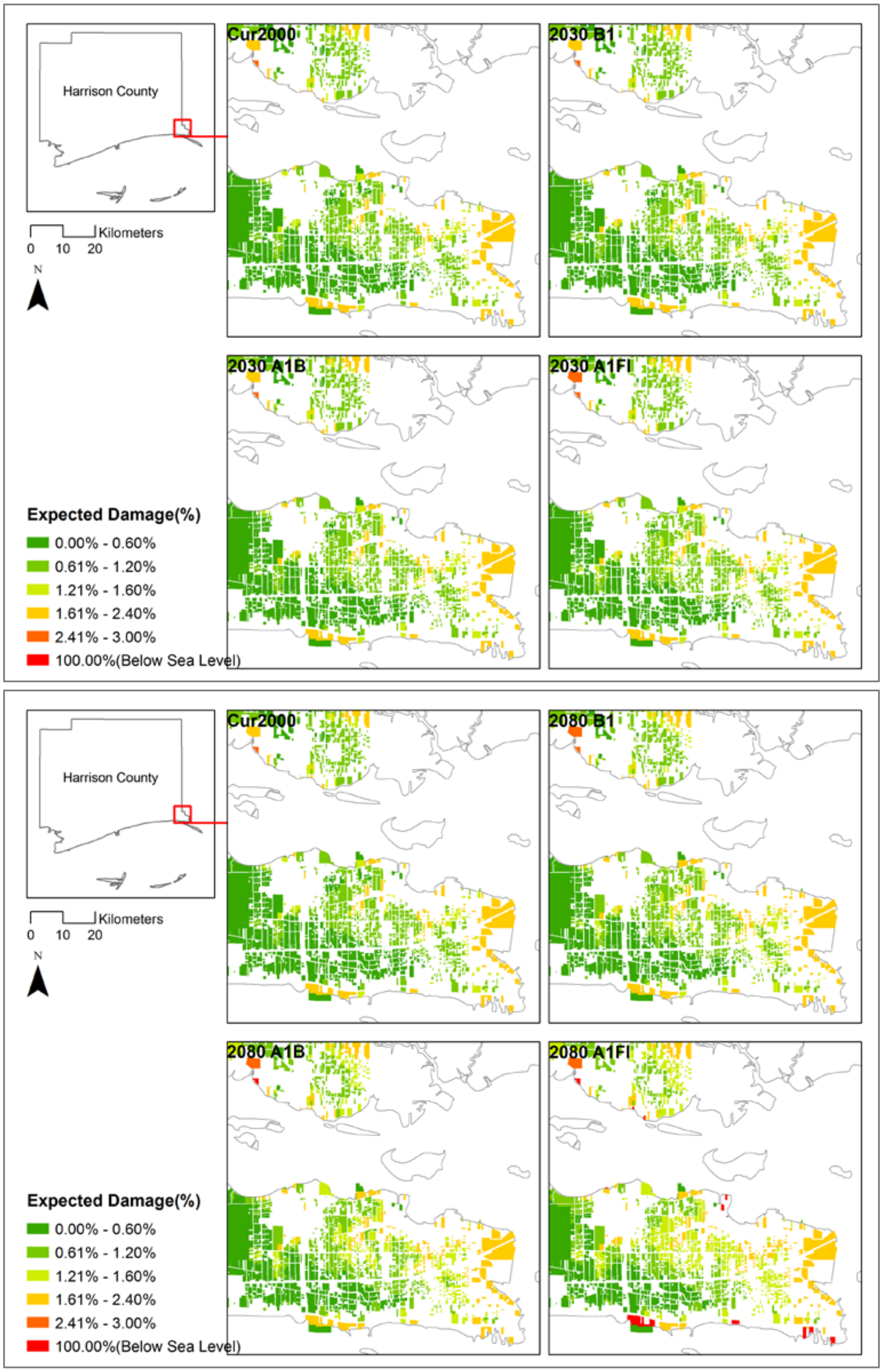


Figure 5-21 Expected Annual Loss at Census Block Level (Biloxi Bay, Gulfport, MS)

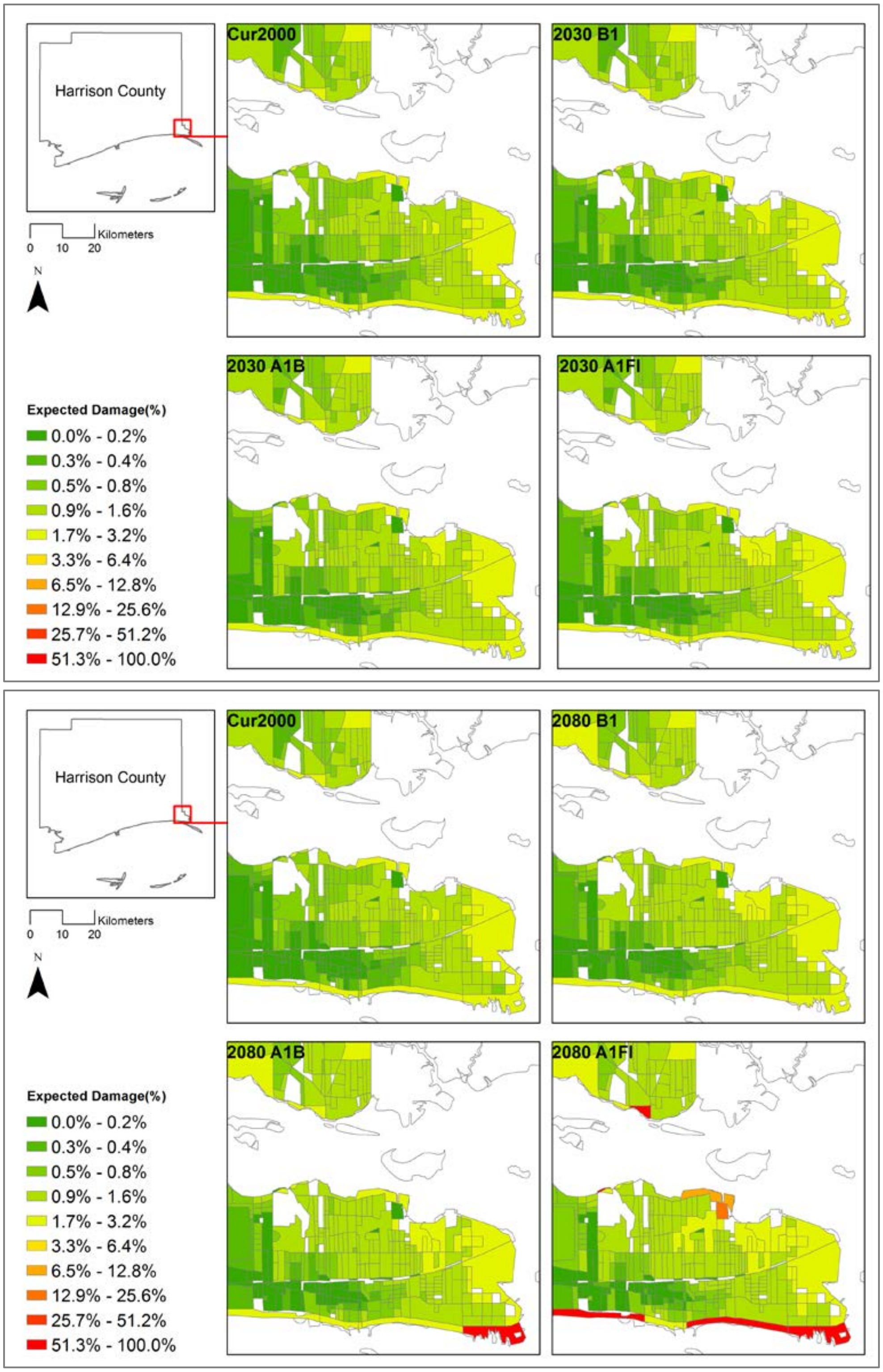


Figure 5-22 Expected Annual Loss at Census Block Level (Biloxi Bay, MS)

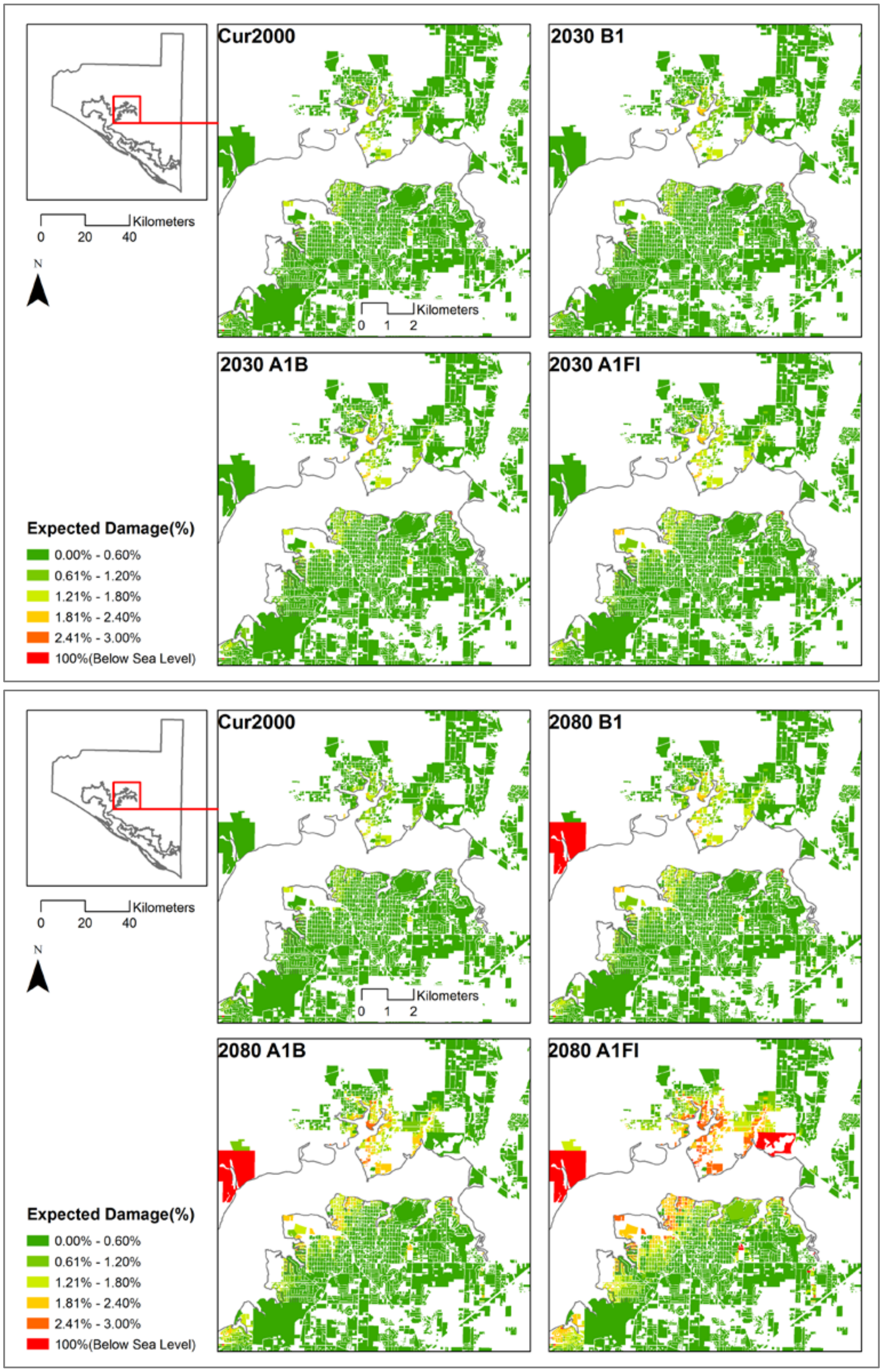


Figure 5-23 Expected Annual Loss at Parcel Level (North Bay, Panama, MS)

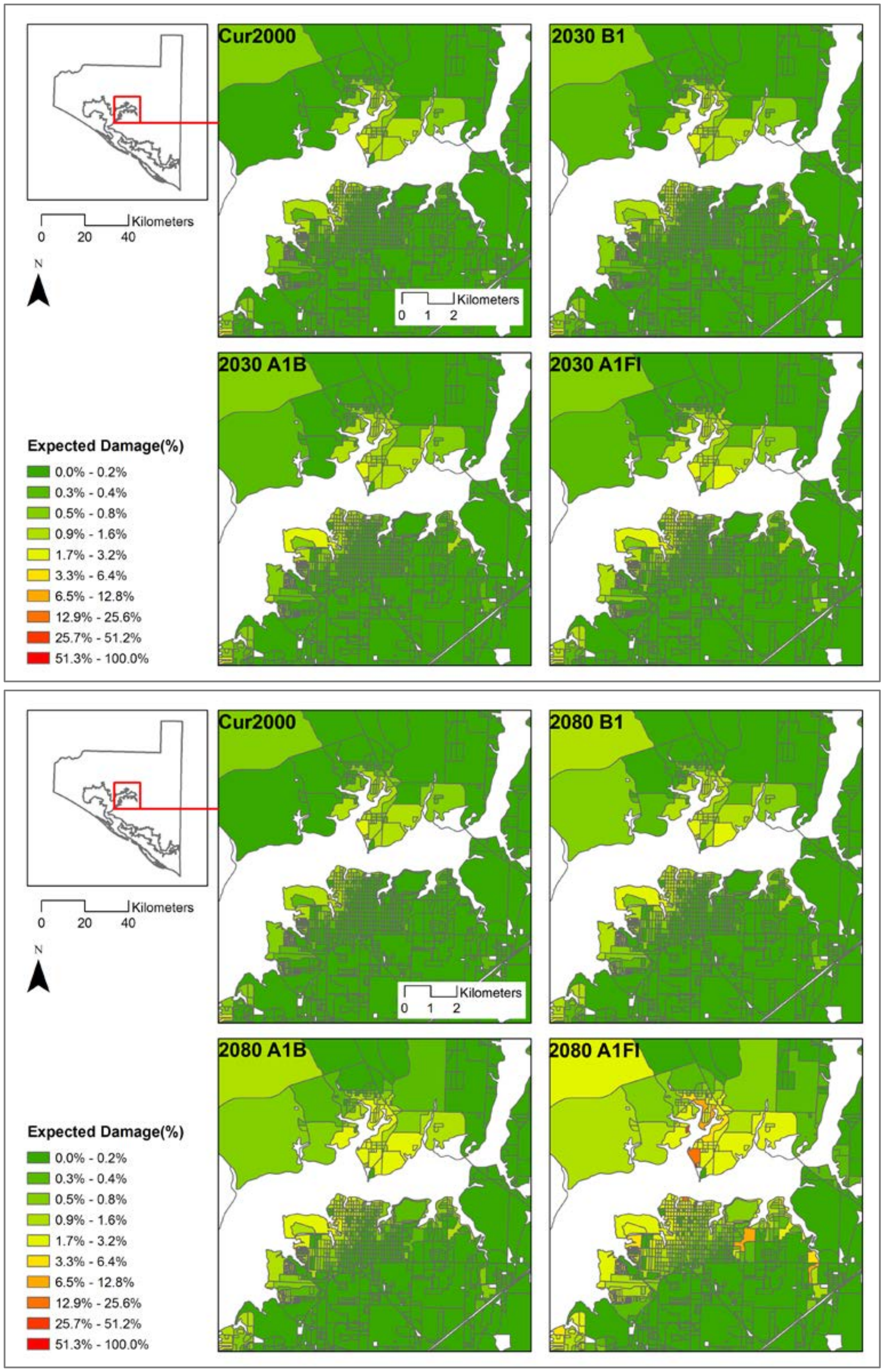


Figure 5-24 Expected Annual Loss at Census Block Level (North Bay, Panama, MS)

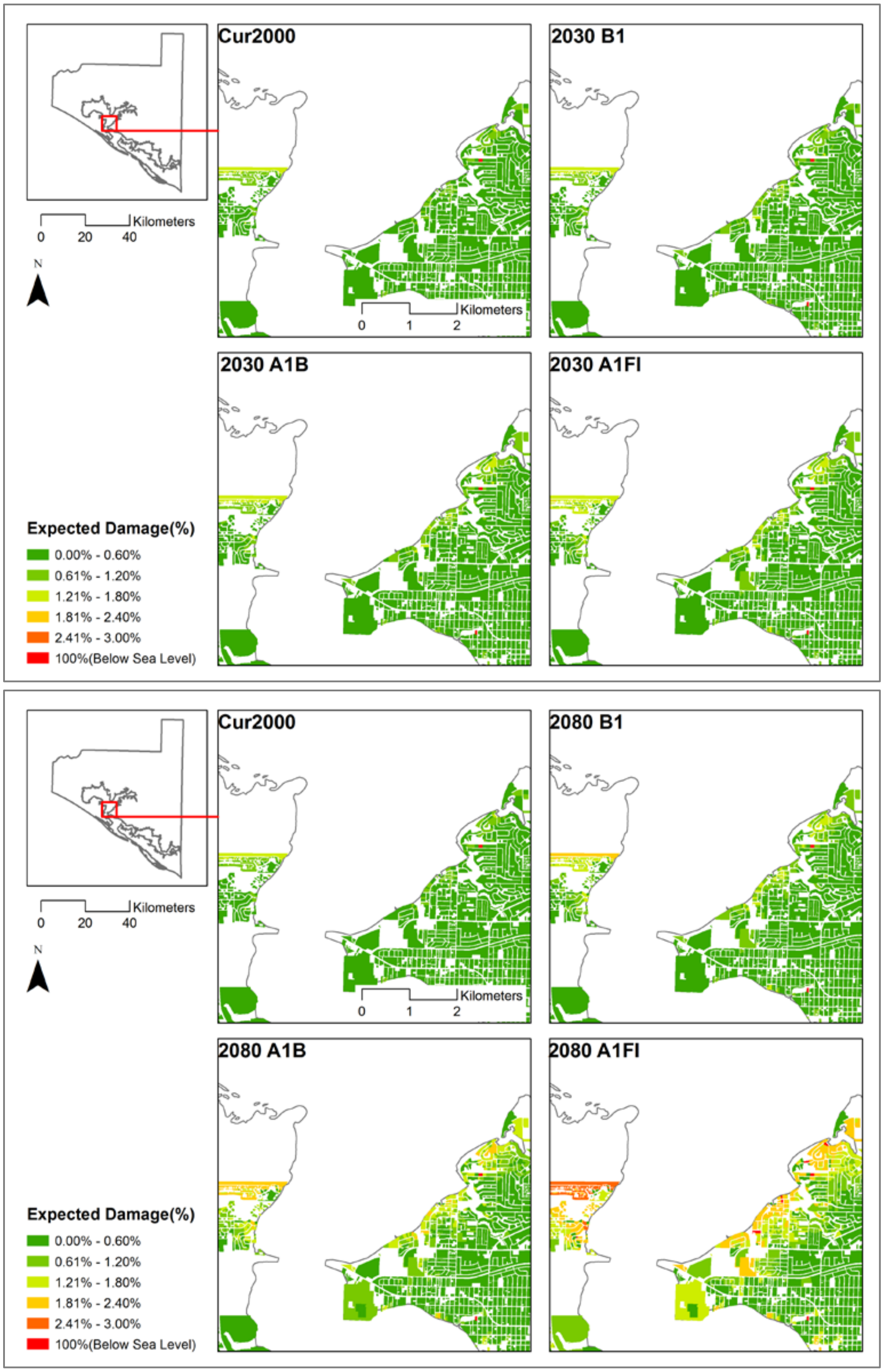


Figure 5-25 Expected Annual Loss at Parcel Level (North Bay, Panama, MS)

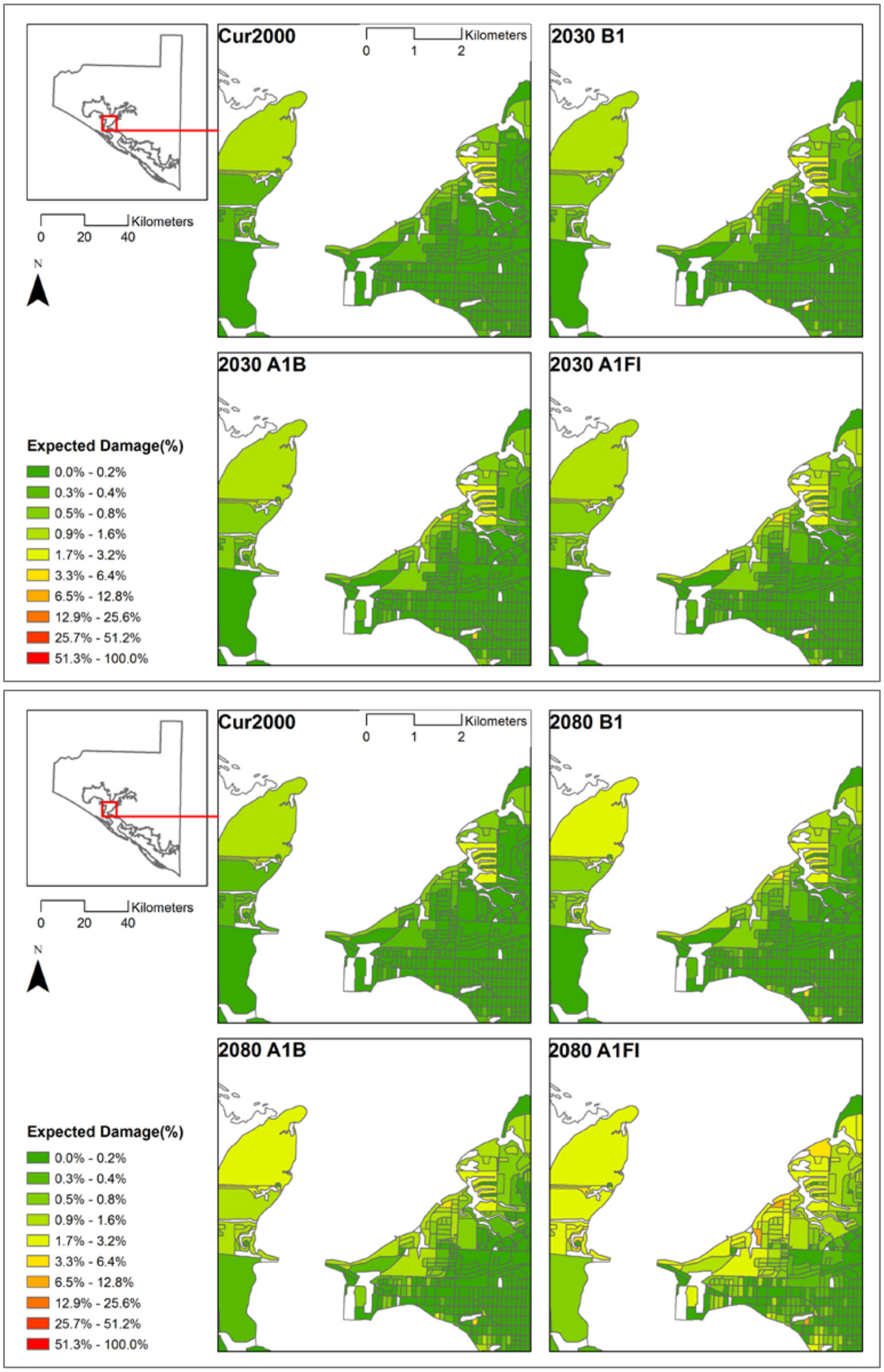


Figure 5-26 Expected Annual Loss at Census Block Level (North Bay, Panama, MS)

5.4. Surge Flood Damage, Affected Population, and Flooded Area for Different Returned Periods

Applying the approach for calculating the Surge Flood Damage, Affected Population, and Flooded Area for different Returned Periods discussed in the Methodology section, the results are shown as Figure 5-27 through Figure 5-32. For example, Figure 5-27 and Figure 5-28 are results for Corpus Christi. Figure 5-27 is the result which consider the properties below future sea level as complete loss, whereas Figure 5-28 is the result that assumes the property below future sea level due do SLR do not exist at selected time points (year 2030 and 2080). The comparison of these two types of results indicates that even if we do not consider the property loss due to SLR, the future climate conditions could still cause a higher amount of damage, affected more people and flood more area, and the value will be significantly greater under the 2080 scenarios than under current climate condition.

The results of Gulfport (Figure 5-29 and Figure 5-30) and Panama (Figure 5-31 and Figure 5-32) show similar patterns as that of Corpus Christi. Note that in Figure 5-30 the damage vs. return period curves for A1B and A1FI in 2080 (the one at bottom left) almost overlaps with each other. This means the damage increase due to climate condition change is about the same amount of the property value below the sea level in 2080.

Under current climate condition (0 SLR and 0 SST increase), a 100 year event could cause damage of 50 million, 700 million, and 200 million dollars; affected population of 8,000, 30,000 and 15,000; flood area of 160, 130 and 180 km^2 for Corpus Christi, Gulfport and Panama City respectively. Under different future climate conditions, the damage, affected population and flooded area would increase by different amount of values for different cities.

If consider the property below future sea level as compete loss. For Corpus Christi under 2030 climate condition, the 100 year event damage could increase by as much as 80 million dollars, while the affected population could increase by as much as 5,000 and the flooded area could increase by as much as 40 km^2. Under 2080 climate condition, the 100 year events flood damage could increase by as much as 330 million dollars, while the affected population could increase by as much as 15,000 and the flooded area could increase by as much as 100 km^2.

For Gulfport under 2030 climate condition, the 100 year event damage could increase by as much as 150 million dollars, while the affected population could increase by as much as 8,000 and the flooded area could increase by as much as 30 km^2. Under 2080 climate conditions, the 100 year events flood damage could increase by as much as 500 million dollars, while the affected population could increase by as much as 30,000 and the flooded area could increase by as much as 100 km^2.

For Panama, if consider the property below future sea level as compete loss, under 2030 climate condition, the 100 year event damage could increase by as much as 100 million dollars, while the affected population could increase by as much as 6,000 and the flooded area could increase by as much as 30 km^2. Under 2080 climate conditions, the 100 year events flood damage could increase by as much as 500 million dollars, while the affected population could increase by as much as 20,000 and the flooded area could increase by as much as 120 km^2.

The comparison of flood damage, affected population and flooded area under current and future climate conditions for three cities, are listed in Table 5–1 through Table 5–3.

Table 5–1 100 Year Event Damage (in Million Dollars) Comparison for Three Cities

	Current	2030	2080
Corpus Christi	50	+30~80	+100~330
Gulfport	700	+70~150	+250~500
Panama	200	+30~100	+100~500

Table 5–2 100 Year Event Affected Population Comparison for Three Cities

	Current	2030	2080
Corpus Christi	8,000	+2500~5,000	+6,000~15,000
Gulfport	30,000	+4,000~8,000	+10,000~30,000
Panama	15,000	+2,000~6,000	+6,500~20,000

Table 5–3 100 Year Event Flooded Area (in km^2) Comparison for Three Cities

	Current	2030	2080
Corpus Christi	160	+20~40	+50~100
Gulfport	130	+15~30	+60~100
Panama	180	+12~30	+40~120

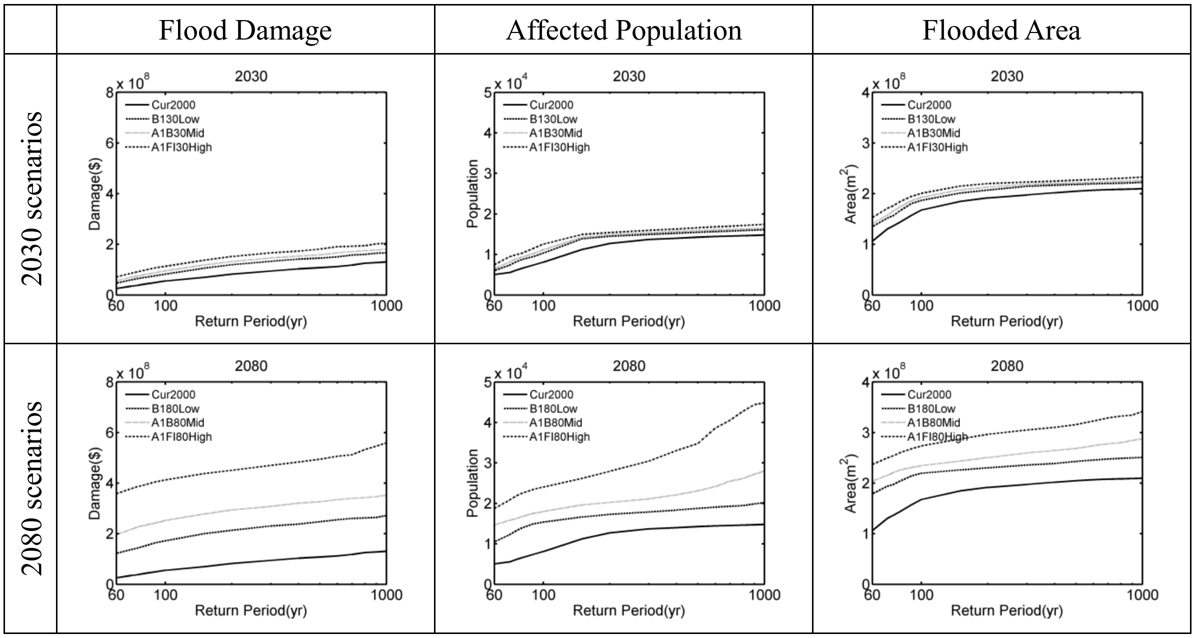


Figure 5-27 The Damage, Affected Population, and Flooded Area for Different Returned Periods in Corpus Christi, TX (Consider the Properties Below Future Sea Level as Complete Loss)

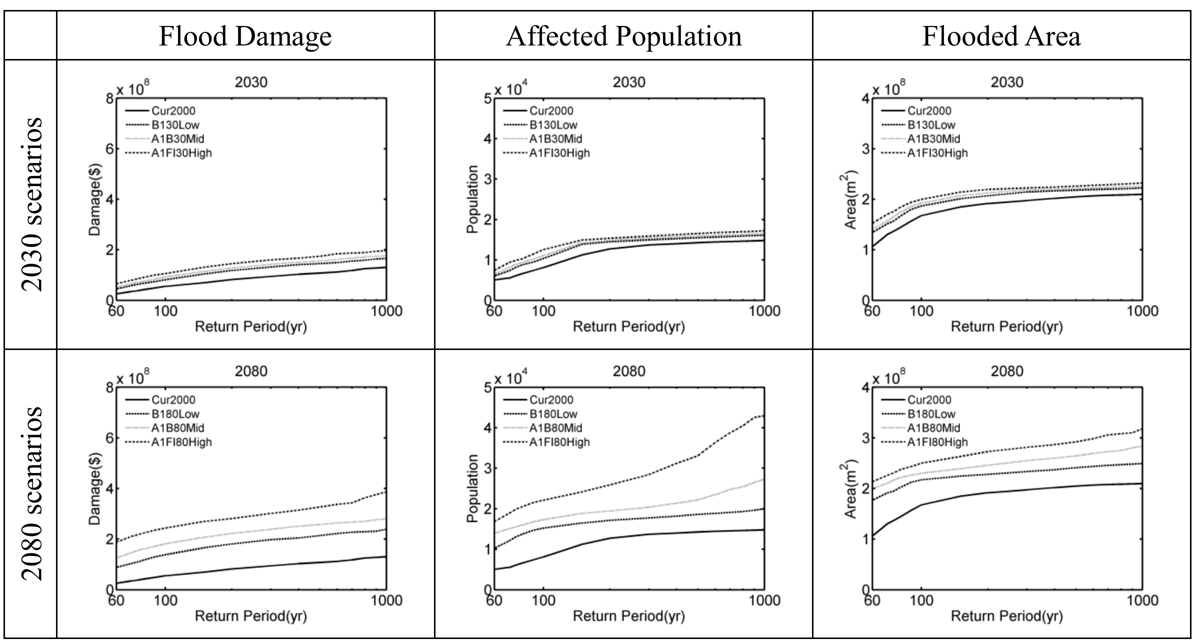


Figure 5-28 The Damage, Affected Population, and Flooded Area for Different Returned Periods in Corpus Christi, TX (Without Considering Properties Below Future Sea Level)

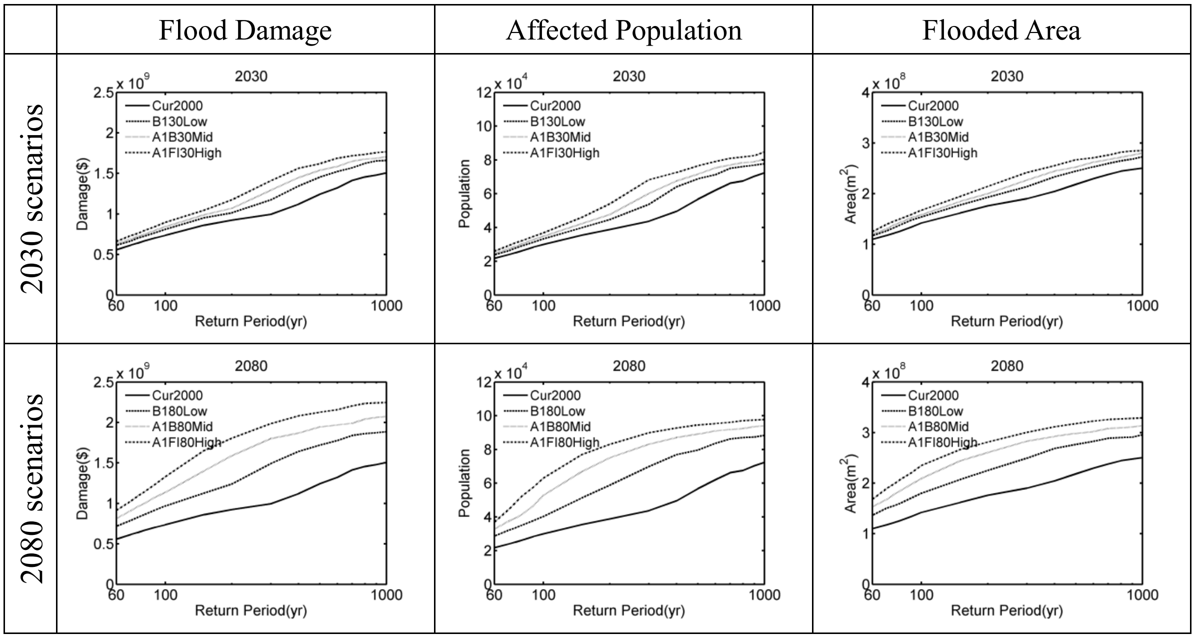


Figure 5-29 The Damage, Affected Population, and Flooded Area for Different Returned Periods in Gulfport, MS (Consider the Properties below Future Sea Level as Complete Loss)

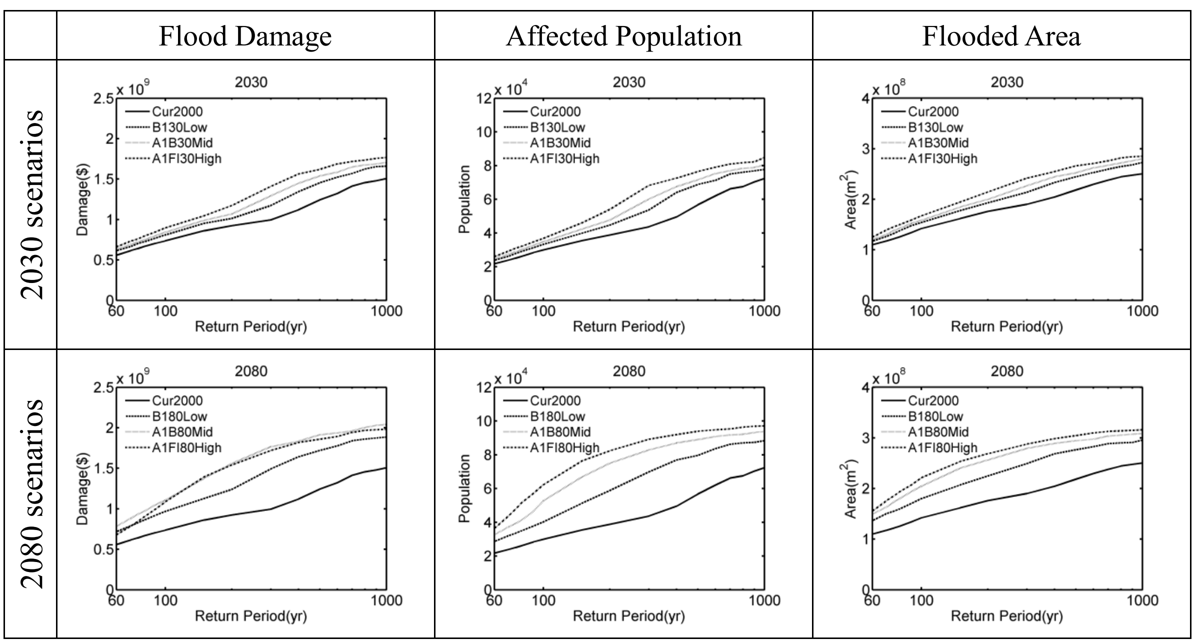


Figure 5-30 The Damage, Affected Population, and Flooded Area for Different Returned Periods in Gulfport, MS (Without Considering Properties below Future Sea Level)

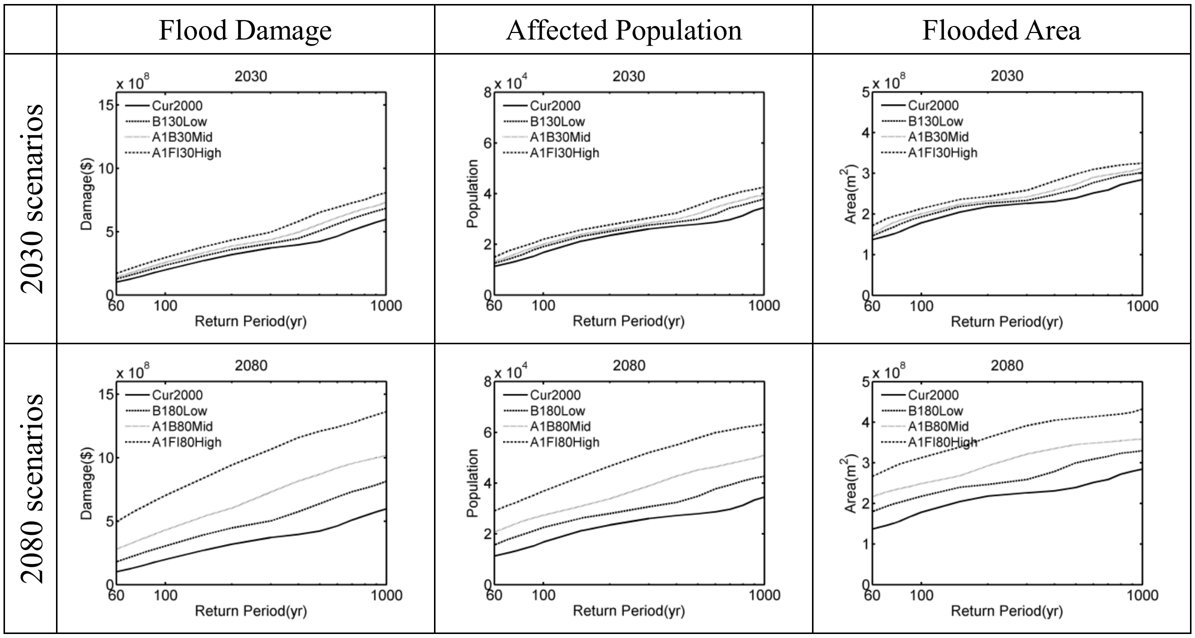


Figure 5-31 The Damage, Affected Population, and Flooded Area for Different Returned Periods in Panama, FL (Consider the Properties below Future Sea Level as Complete Loss)

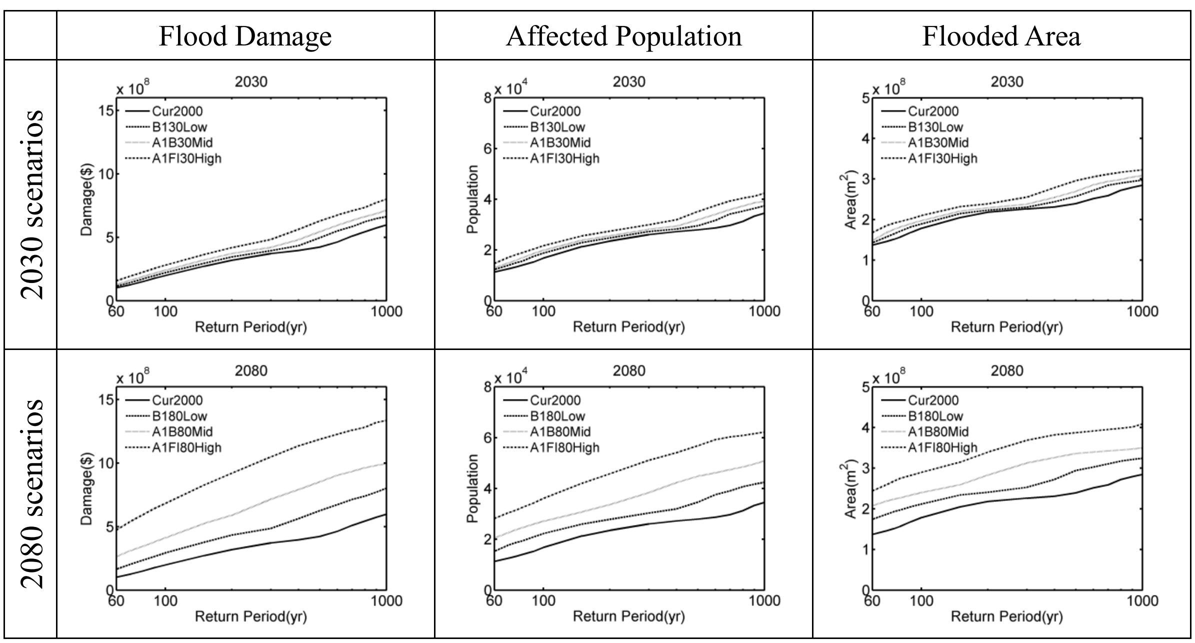


Figure 5-32 The Damage, Affected Population, and Flooded Area for Different Returned Periods in Panama, FL (Without Considering Properties below Future Sea Level)

5.5. Affected Business, Number of Employee and Sales Volume for Different Returned Periods

The affected business (i.e. sales volume, number of employee and number of business) vs. return period relationships are developed for the business at three study areas (Figure 5-33 through Figure 5-38). Under current climate condition, a 100 year event could affect 500, 2,100, and 700 businesses; affected 4,000, 30,000 and 4,000 employee; affect 500, 4,000 and 700 million dollars of sales volume for Corpus Christi, Gulfport and Panama City respectively. Similar to the analysis done for parcel data, under different future climate conditions, the affected business, employee and sales volume would increase by different amount of values for different cities. The comparison of affected number of businesses, employee and sales volume under current and future climate conditions for three cities, are listed in Table 5–4 through Table 5–6.

Table 5–4 100 Year Event Affected Business Comparison for Three Cities

	Current	2030	2080
Corpus Christi	500	+400~800	+900~1,400
Gulfport	2,100	+300~1,000	+1,000~2,400
Panama	700	+100~300	+200~1,100

Table 5–5 100 Year Event Affected Employee Comparison for Three Cities

	Current	2030	2080
Corpus Christi	4,000	+4,000~8,000	+9,000~13,000
Gulfport	30,000	+3,000~7,000	+10,000~30,000
Panama	4,000	+800~2,000	+2,000~10,000

Table 5–6 100 Year Event Affected Sales Volume Comparison for Three Cities

(in Million Dollars)	Current	2030	2080
Corpus Christi	500	+500~1,000	+1,100~1,600
Gulfport	4,000	+500~1,500	+1,600~4,000
Panama	700	+100~300	+200~1,400

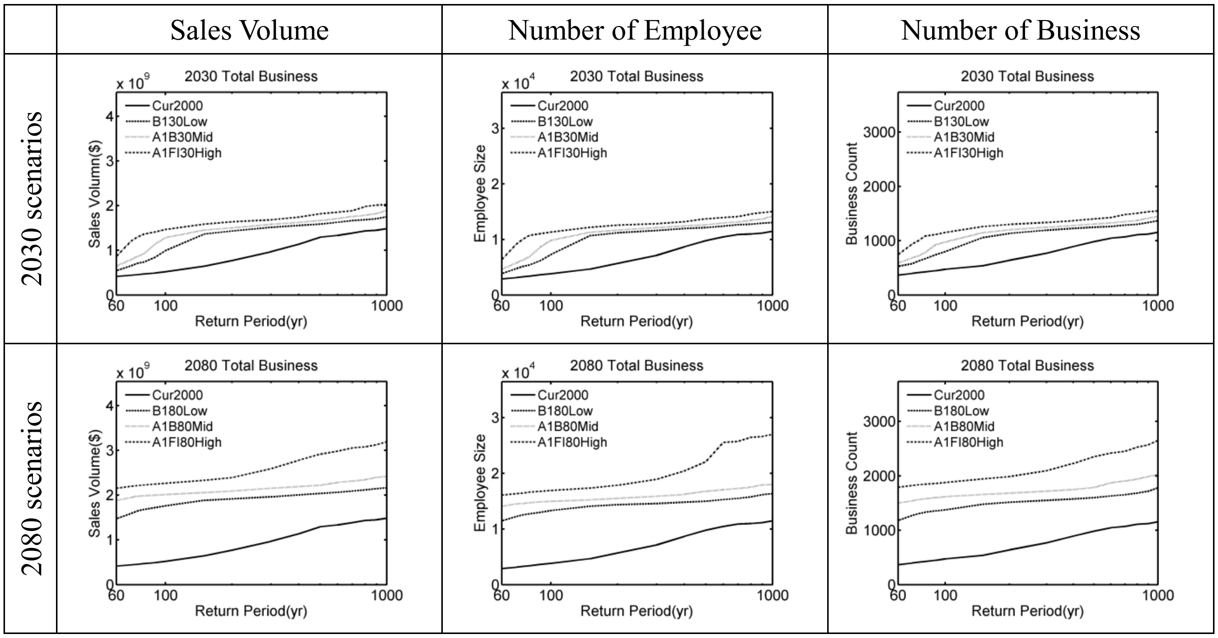


Figure 5-33 The Sales Volume, Number of Employee and Number of Business for Different Returned Periods In Corpus Christi, TX (Consider the Business below Future Sea Level as Complete Loss)

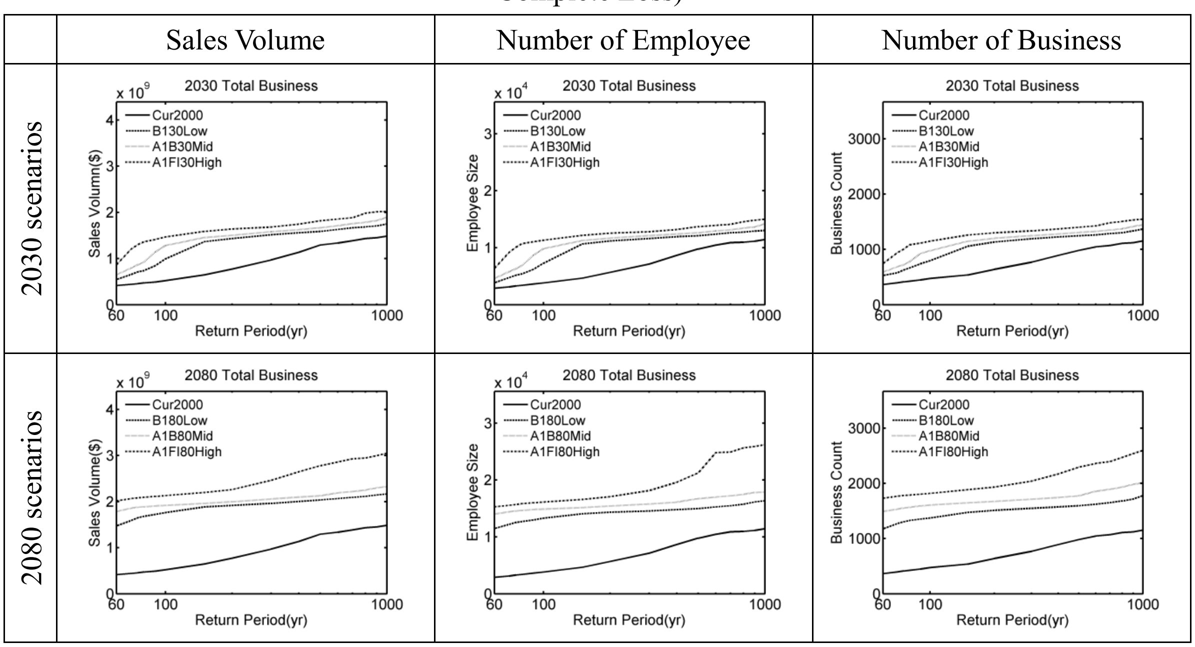


Figure 5-34 The Sales Volume, Number of Employee and Number of Business for Different Returned Periods In Corpus Christi, TX (Without Considering Properties below Future Sea Level)

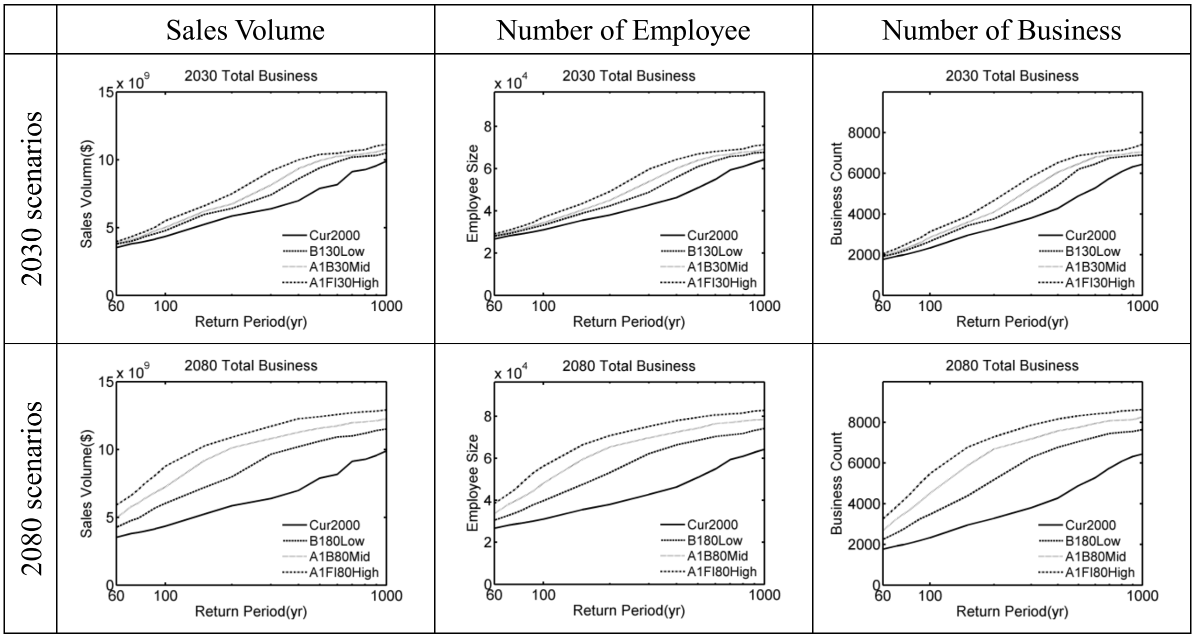


Figure 5-35 The Sales Volume, Number of Employee and Number of Business for Different Returned Periods In Gulfport, MS (Consider the Business Below Future Sea Level as Complete Loss)

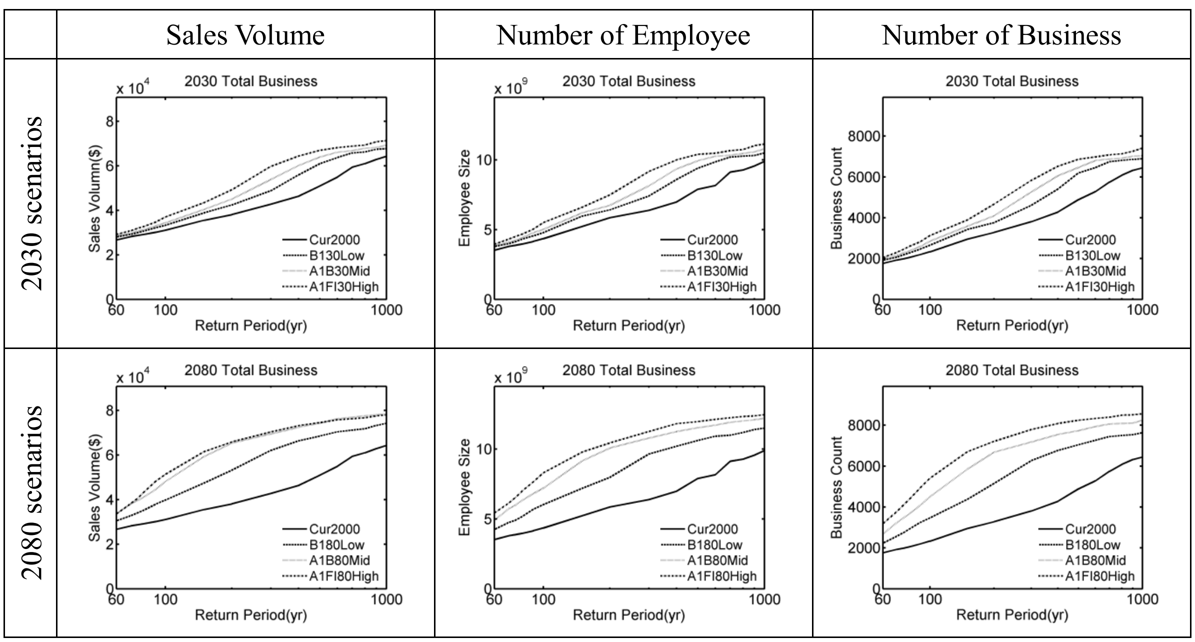


Figure 5-36 The Sales Volume, Number of Employee and Number of Business for Different Returned Periods In Gulfport, MS (Without Considering Properties below Future Sea Level)

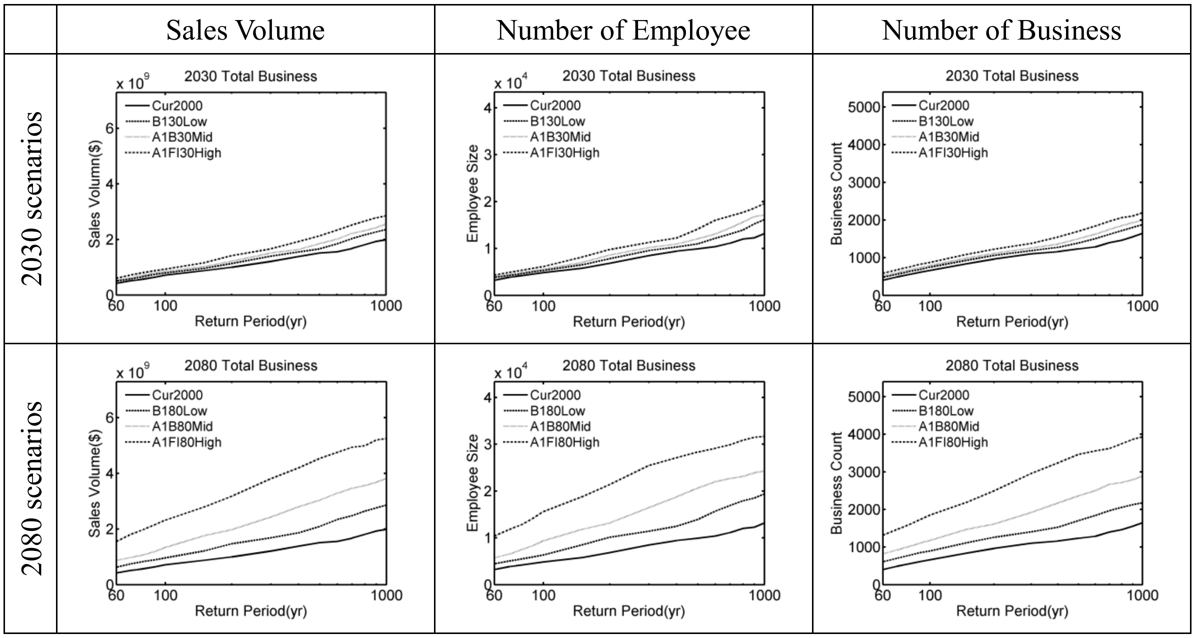


Figure 5-37 The Sales Volume, Number of Employee and Number of Business for Different Returned Periods In Panama, FL (Consider the Business below Future Sea Level as Complete Loss)

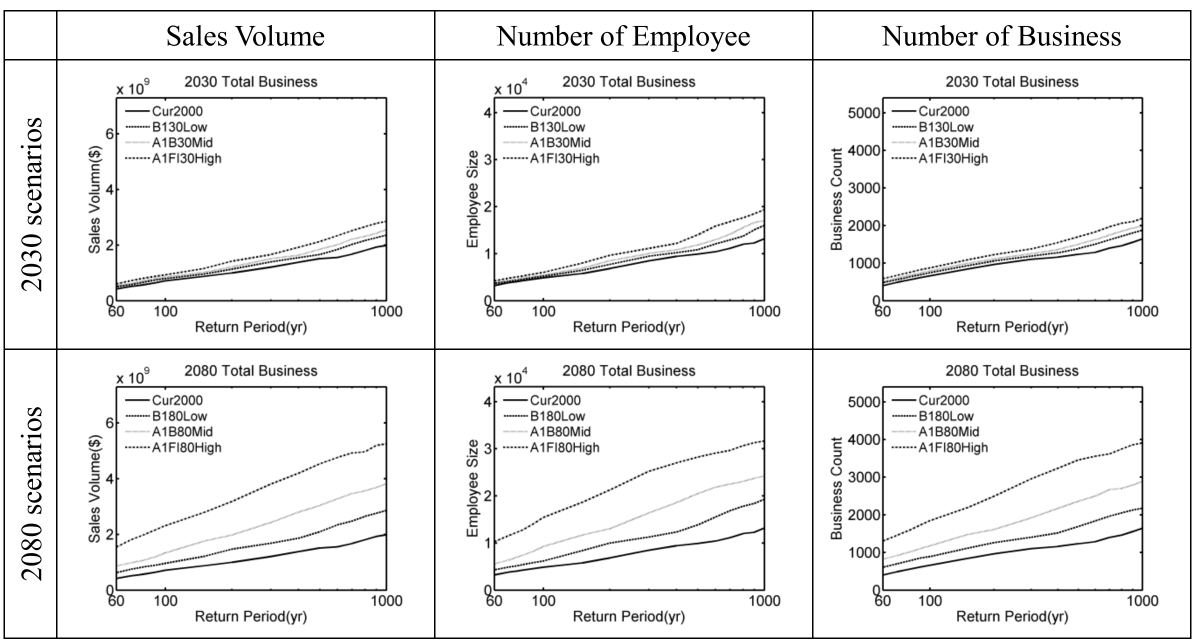


Figure 5-38 The Sales Volume, Number of Employee and Number of Business for Different Returned Periods In Panama, FL (Without Considering Properties below Future Sea Level)

The analysis for business is also conducted for the top four businesses (ranked by number of employee) for each study area (See Appendix G for complete results). Most of the results show similar patterns. One of the cases worth noting is the result for the rank 3 business (Ambulatory Health Care Services) in Corpus Christi (see Figure 5-39 and Figure 5-40). The sudden jump of the curves for the 2080 scenarios indicate that some companies locate at high elevation locations and account for a relatively large amount of sales volume (and number of employee) comparing to the rest of the companies. Hence when the return period exceeds a certain threshold (which means the surge exceeds a certain threshold), these companies at high elevation locations are flooded and the affected amount of sales volume and number of employ are reflected on the curves.

Another special case is the result for the rank 1 business (Amusement, Gambling, and Recreation Industries business) in Gulfport (Figure 5-41 and Figure 5-42). The reason why the change of values is more significant in the number of affected business than the sales volume and number of employee is that, few companies account for the majority of sales volume and number of employee and get flooded for events with small return period. Hence when the number of affected companies increase as the return period gets greater, the change of value on sales volume and number of employee are relatively small.

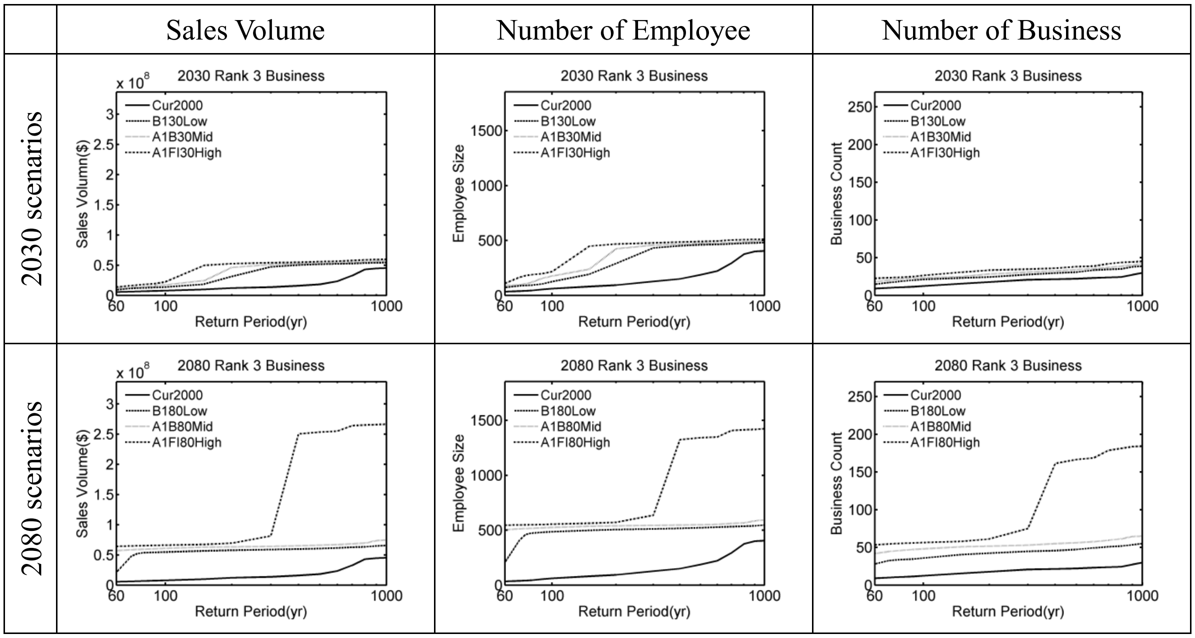


Figure 5-39 Corpus Christi Rank 3 Business: Ambulatory Health Care Services (Consider the Business below Future Sea Level as Complete Loss)

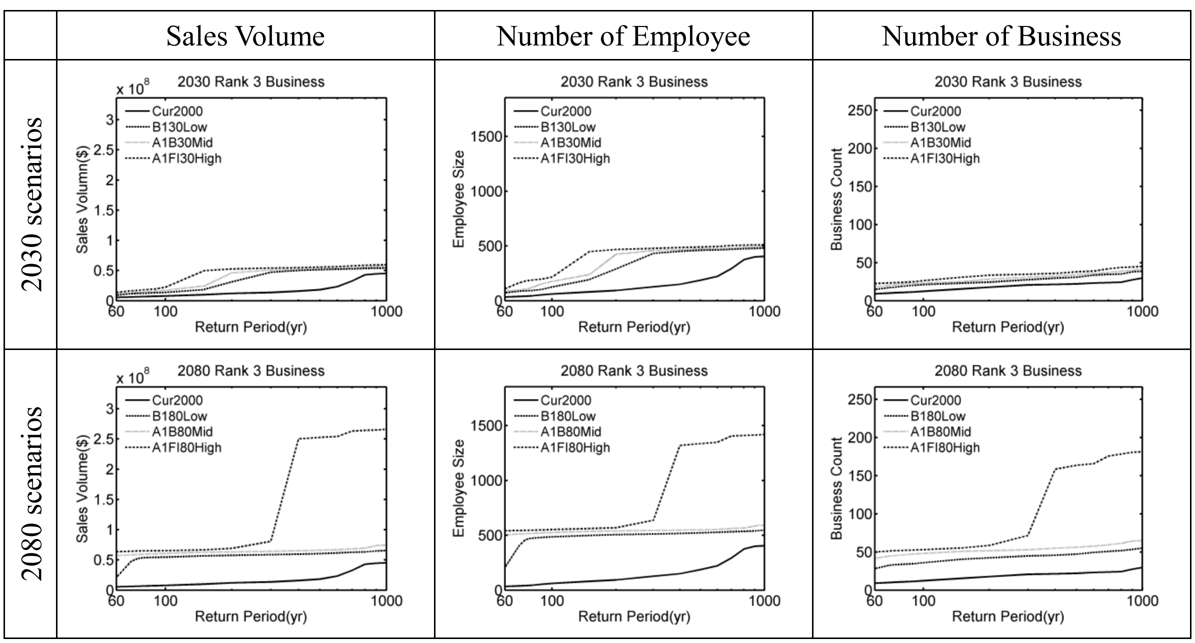


Figure 5-40 Corpus Christi Rank 3 Business: Ambulatory Health Care Services (Without Considering Properties below Future Sea Level)

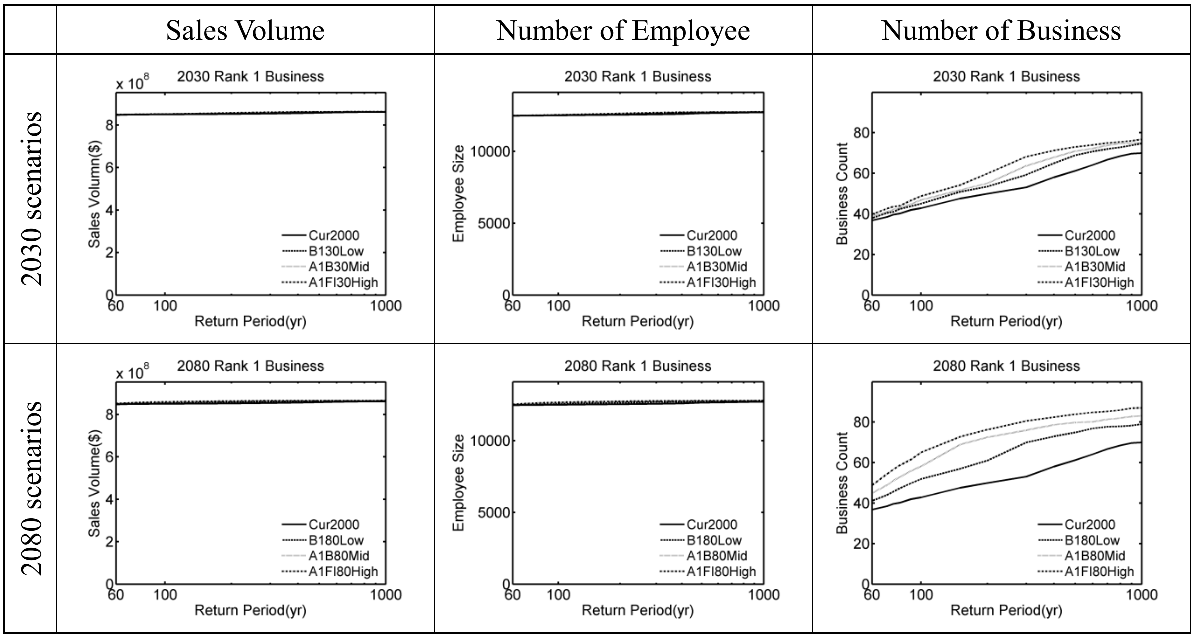


Figure 5-41 Gulfport Rank 1 Business: Amusement, Gambling, and Recreation Industries (Consider the Business below Future Sea Level as Complete Loss)

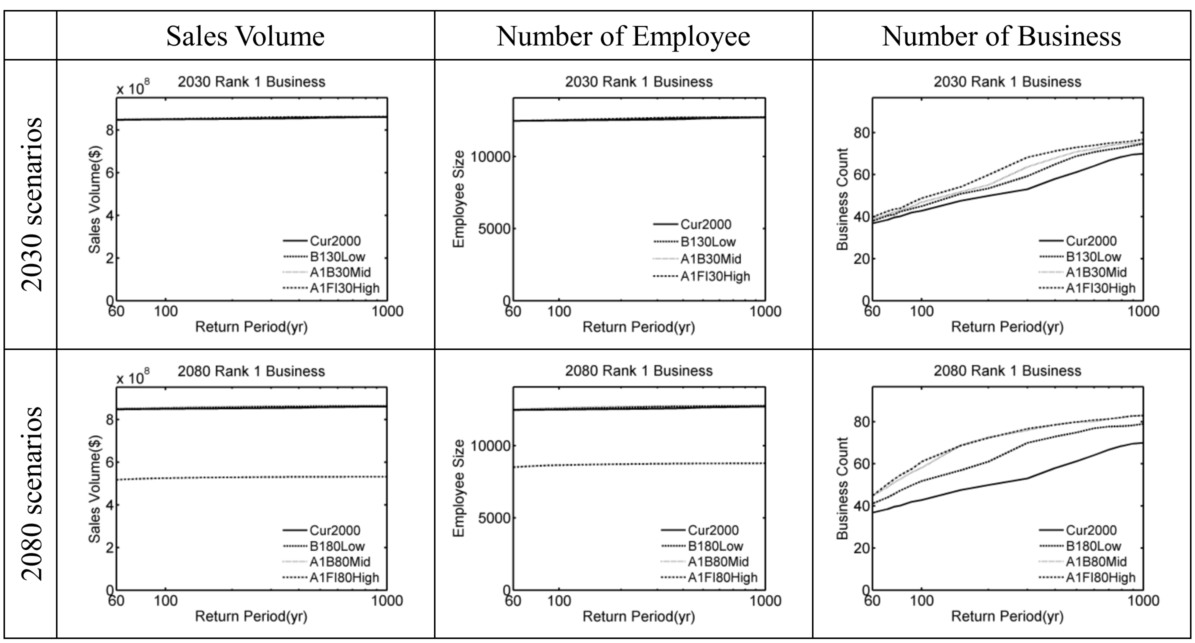


Figure 5-42 Gulfport Rank 1 business: Amusement, Gambling, and Recreation Industries (Without Considering Properties below Future Sea Level)

5.6. Effect of Hurricane Parameters on Hurricane Flood Damage Estimation

The Surge Response Function allows us to obtain the flood damage, affected population, flooded area and affected business in an efficient manner. When implementing, all the values of interest (the flood damage, affected population and flooded area) can be pre-calculated and store in tables. Table 5–7 gives the idea that all the values of interest are function of hurricane parameters. With tables like Table 5–7 we can compare the damage caused by hurricanes with different parameters. Here we demonstrate the application of this "tabular approach" by analyzing the damage caused by a hypothetical hurricane with central pressure of 950 mb and radius of maximum of 16 nm.

Following analysis includes the damage estimation (along with affected population and flooded area) for this hypothetical hurricane making landfall at different locations at the three cities. The future climate conditions are included to investigate the effects of SST and SLR on the damage values. The effects of hurricane parameters on flood damage are explored by calculating the damage value under current climate condition, with one parameters value varies and keep others parameters values fixed.

5.6.1. Landfall Locations

Figure 5-46 shows the flood damage, affected population and flooded area caused by the hypothetical making landfall at different locations at Corpus Christ coast. Similarly, Figure 5-47 and Figure 5-48 show the results for Gulfport and Panama City. Under current climate condition, the flood damage peaks at 100 million, 800 million and 170 million at Corpus Christ, Gulfport and Panama, respectively. Similarly, the affected population peaks at 1,000, 3,000 and 1,200; the flooded area peaks at 250 km^2, 180 km^2 and 110 km^2 at Corpus Christ, Gulfport and Panama, respectively.

Figure 5-46 through Figure 5-48 show the result of future climate conditions on damage estimation for the case mentioned above. The results show that the climate conditions in year 2030 would cause damage (including affected population and flooded area) increase by 10 to 30 % (except for the flood damage in Corpus Christi, which increases up to 100 %), while the conditions in 2080 dramatically increase the damage by 50% to 10 times. This indicates that, while Gulfport is subject to the greatest amount of damage under current climate condition, the future climate conditions could cause Corpus Christi the same amount of damage as Gulfport has.

Table 5–7 Example of Values of Interest as Function of Hurricane Parameters

c_p	R_p	v_f	θ	x_0	Flood Damage (dollar)	Affected population	Flooded area (m^2)
770	4	3	-80	-80	12,798,698	2686	73,149,983
770	4	3	-80	-75	13,574,532	2875	75,803,335
770	4	3	-80	-70	14,389,601	3052	78,106,495
770	4	3	-80	-65	15,396,686	3210	79,138,502
770	4	3	-80	-60	16,464,989	3404	79,981,404
770	4	3	-80	-55	17,375,515	3605	80,669,360
770	4	3	-80	-50	18,325,470	3826	81,736,488
770	4	3	-80	-45	19,380,118	4061	82,859,232
770	4	3	-80	-40	20,580,209	4313	84,087,588
770	4	3	-80	-35	21,993,179	4562	85,455,702
continue	…	…	…	…	…	..	…

Note. c_p: Hurricane central pressure; R_p: Hurricane radius to the maximum wind; v_f: Hurricane forward speed; θ: Hurricane approach angle with respect to the shoreline.

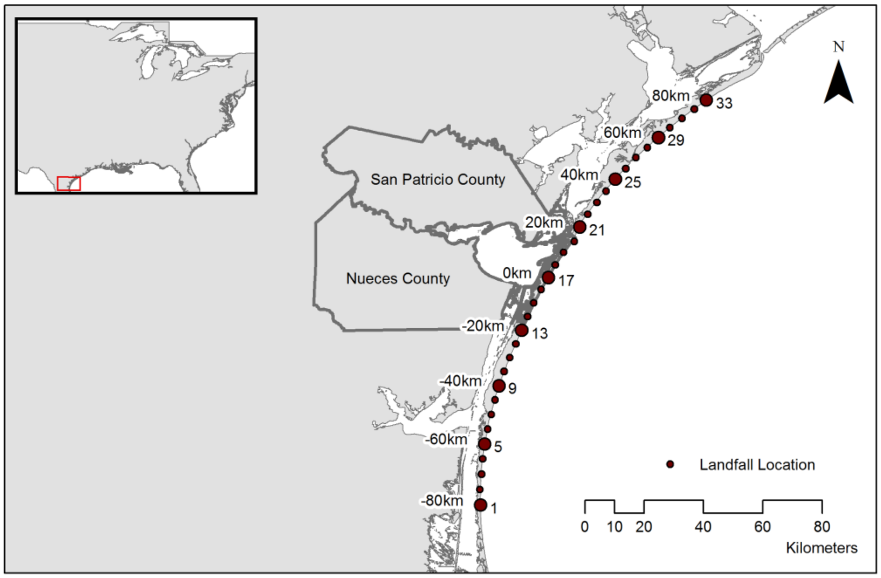


Figure 5-43 Corpus Christi, TX Landfall Locations

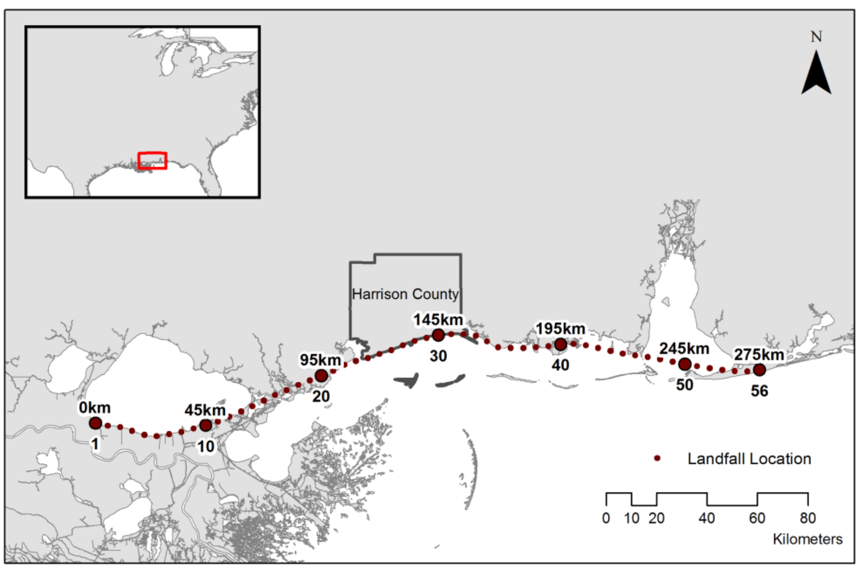


Figure 5-44 Gulfport, MS Landfall Locations

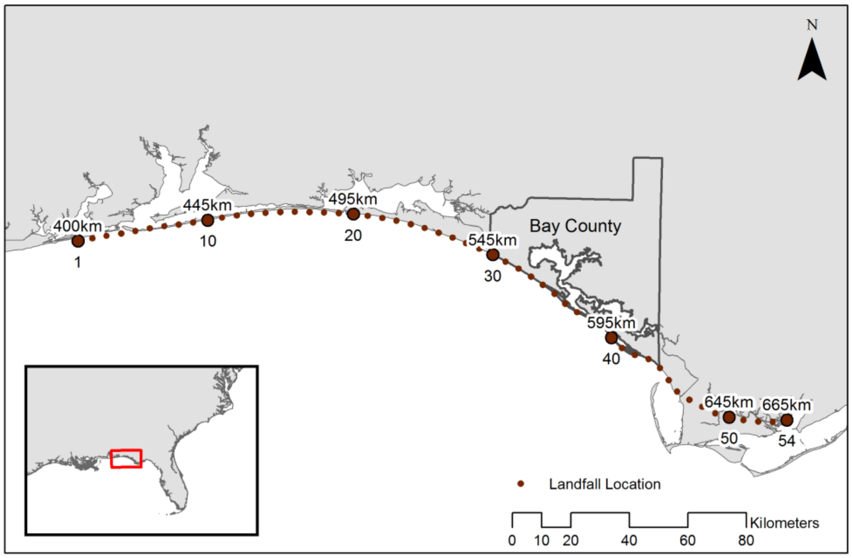


Figure 5-45 Panama, FL Landfall Locations

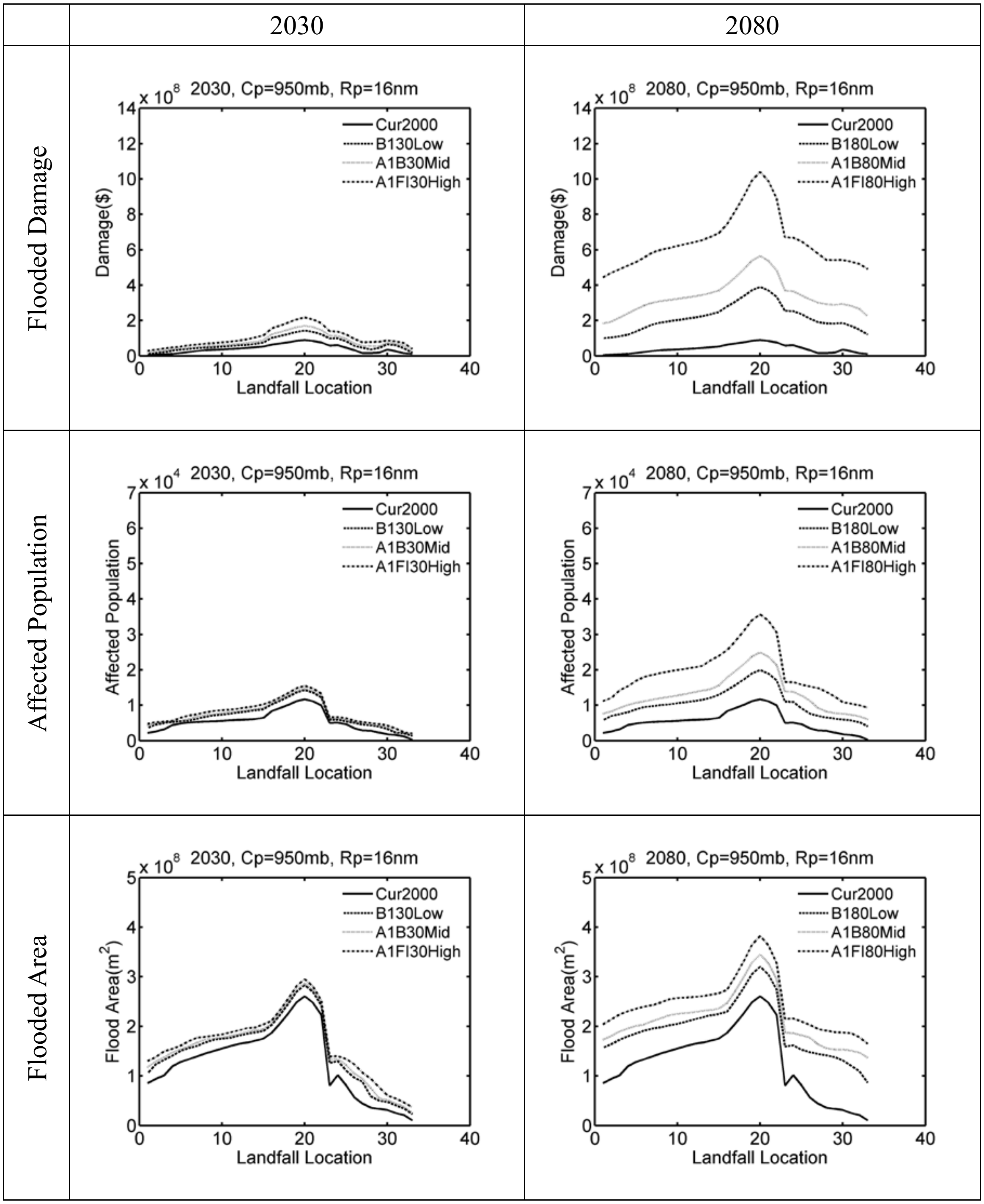


Figure 5-46 Results for Corpus Christi, TX

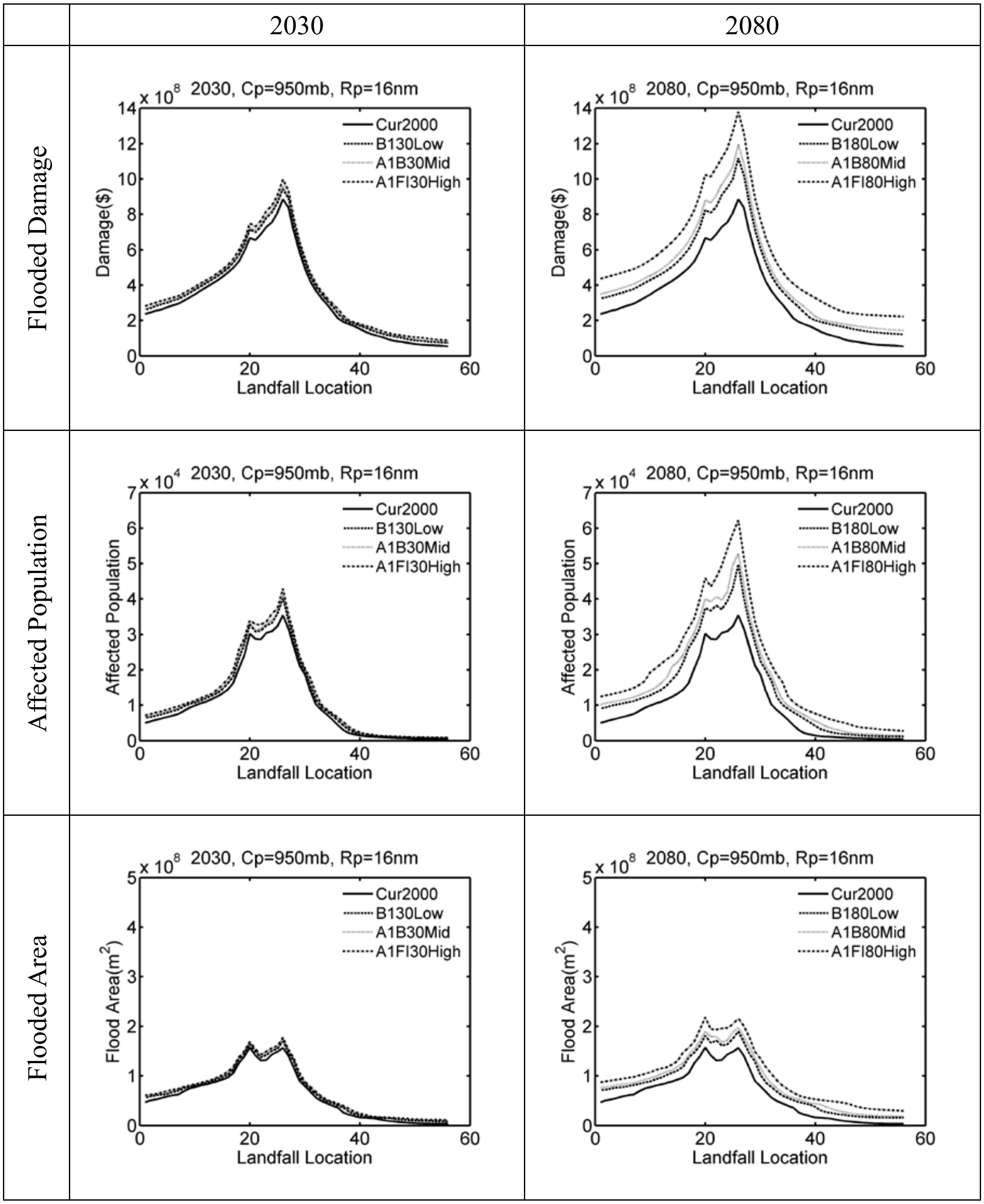


Figure 5-47 Results for Gulfport, MS

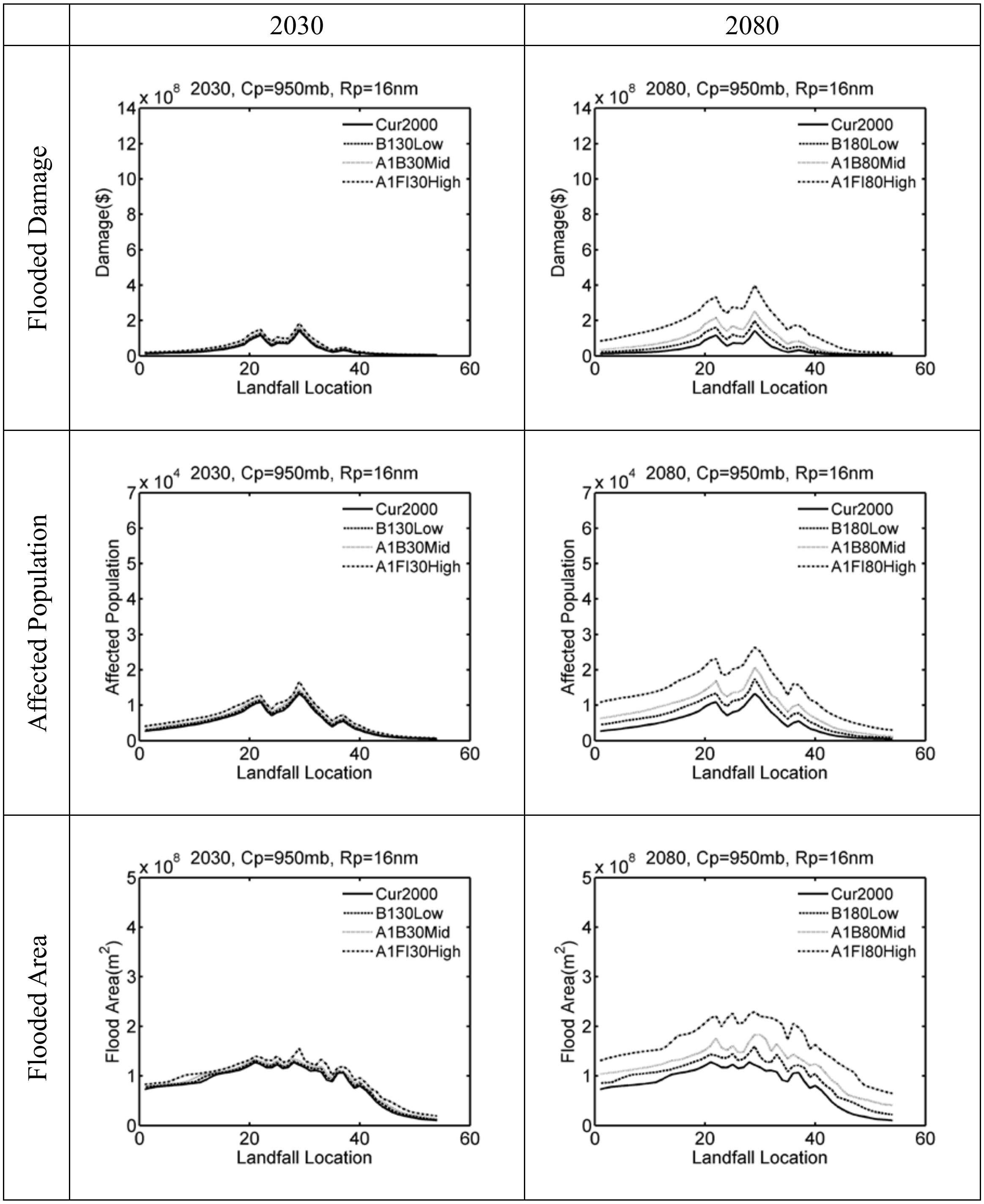


Figure 5-48 Results for Panama, FL

5.6.2. Central Pressure, Radius to the Maximum Wind, Approach Angle and Forward Speed

Figure 5-49 through Figure 5-54 show the effects of central pressure, radius to the maximum wind, approach angle and forward speed on damage estimation under current climate conditions. Note that for the obtained SRF surge data, only Corpus Christi and Panama City include approach angle and only Corpus Christi includes forward speed as hurricane parameters, therefore there are only plots of approach angle effect for two cities and only plots of forward speed effect for one city. The central pressure effect plots (Figure 5-49 and Figure 5-50) show that, with the increase or decrease of 20 mb on central pressure, the damage could reduce to half or increase by 1.5 to 2 times. The radius to the maximum wind effect plots (Figure 5-51 and Figure 5-52) show that the patterns of the curves have changed with regard to different size of radius to the maximum wind. The peak values have changed and the landfall location where the worst scenario occurred has shifted. This is due to the nature of the hurricane wind field structure, i.e. when hurricanes with different radius to the maximum wind make landfall at the same location, the maximum wind actually occur at different locations. Though the shapes of these curves have changed, the results do show that the increase of radius has positive contribution to damage value, while smaller radius results in less damage. The approach angle effect plots (Figure 5-53) show that, with the increase or decrease of 10 degree on approach angle, the damage value would change a little comparing to the results with 0 degree. The negative angle has positive contribution to the damage value, while positive angle results in less damage. The forward speed effect plots (Figure 5-54) show that, with the increase of forward speed, the damage value would increase as well.

The increment of 3 knots (~1.5 m/s) would contribute to approximately 20 million dollars of damage, 1,000 people and 10 km^2 of flooded area.

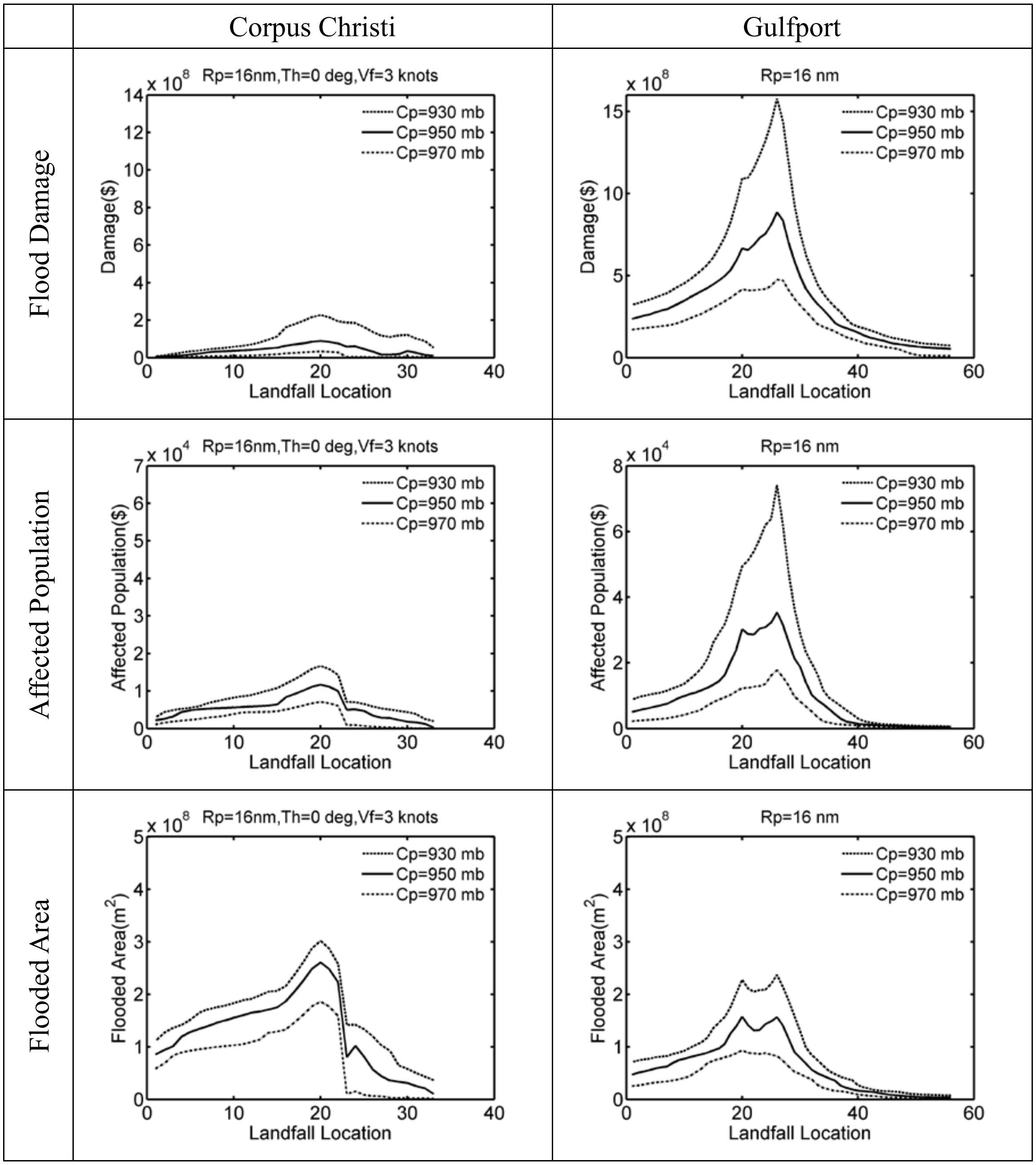


Figure 5-49 The Effect of Central Pressure on Damage Estimation (Corpus Christi and Gulfport)

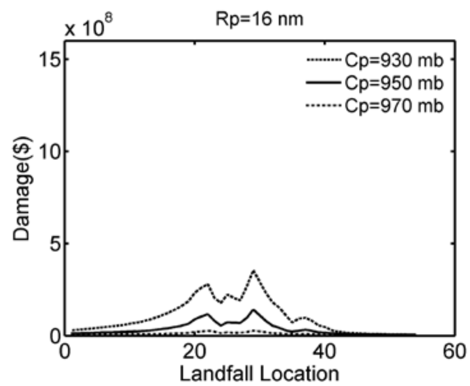
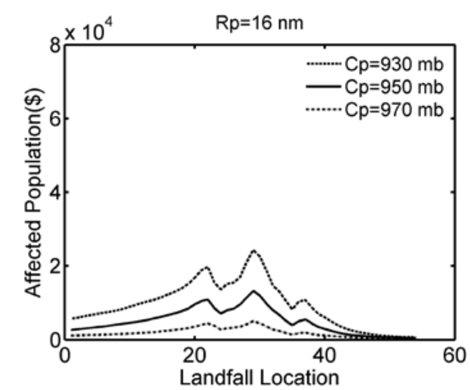
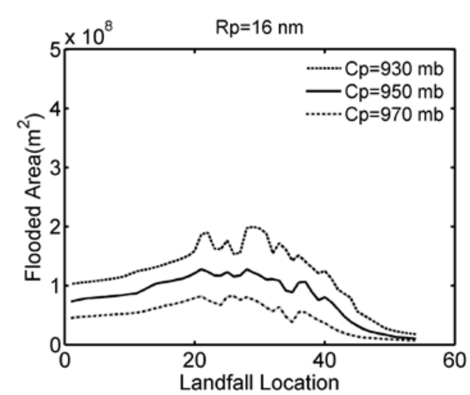
	Panama
Flood Damage	
Affected Population	
Flooded Area	

Figure 5-50 The Effect of Central Pressure on Damage Estimation (Panama)

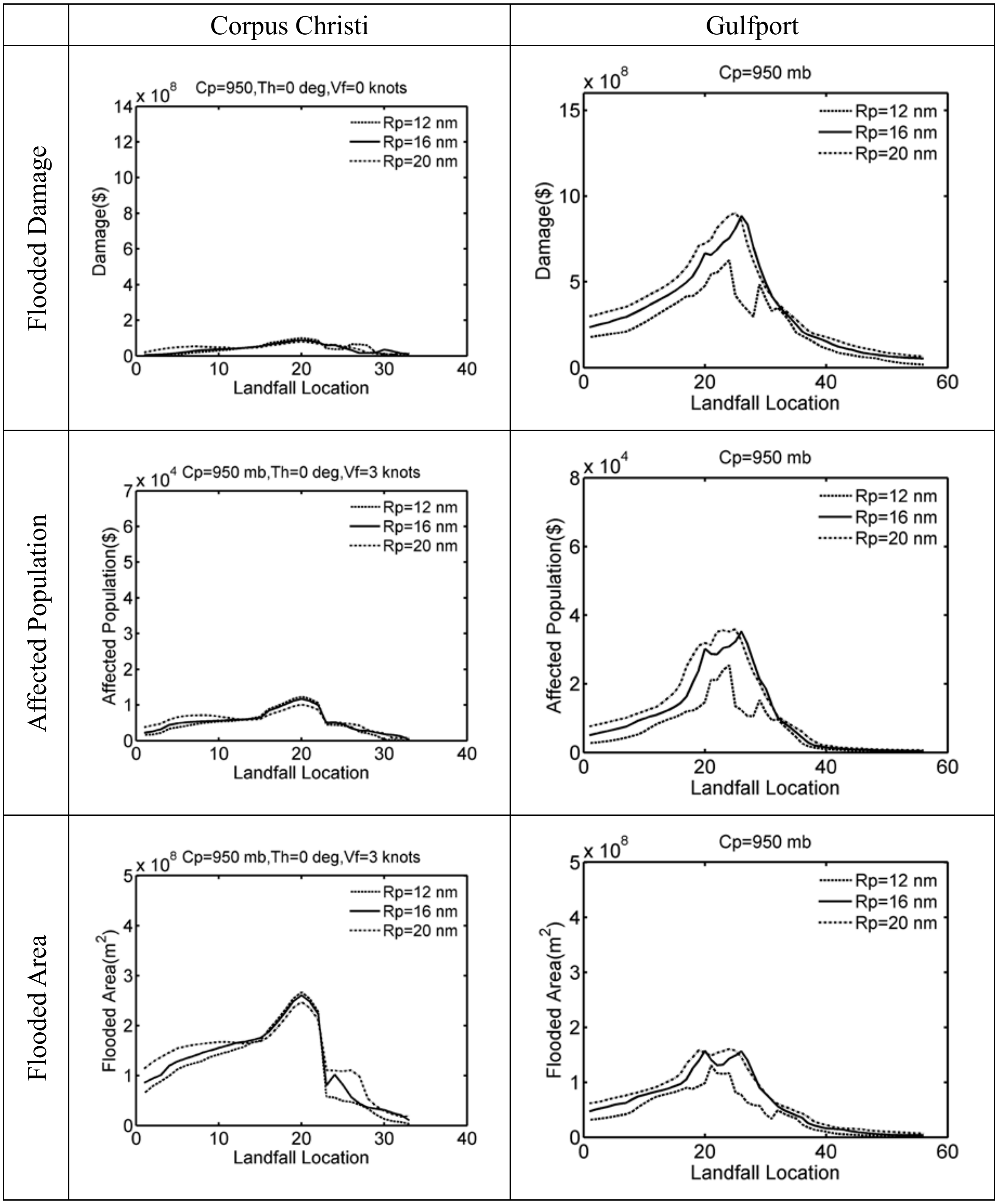


Figure 5-51 The Effect of Radius to the Maximum Wind on Damage Estimation (Corpus Christi and Gulfport)

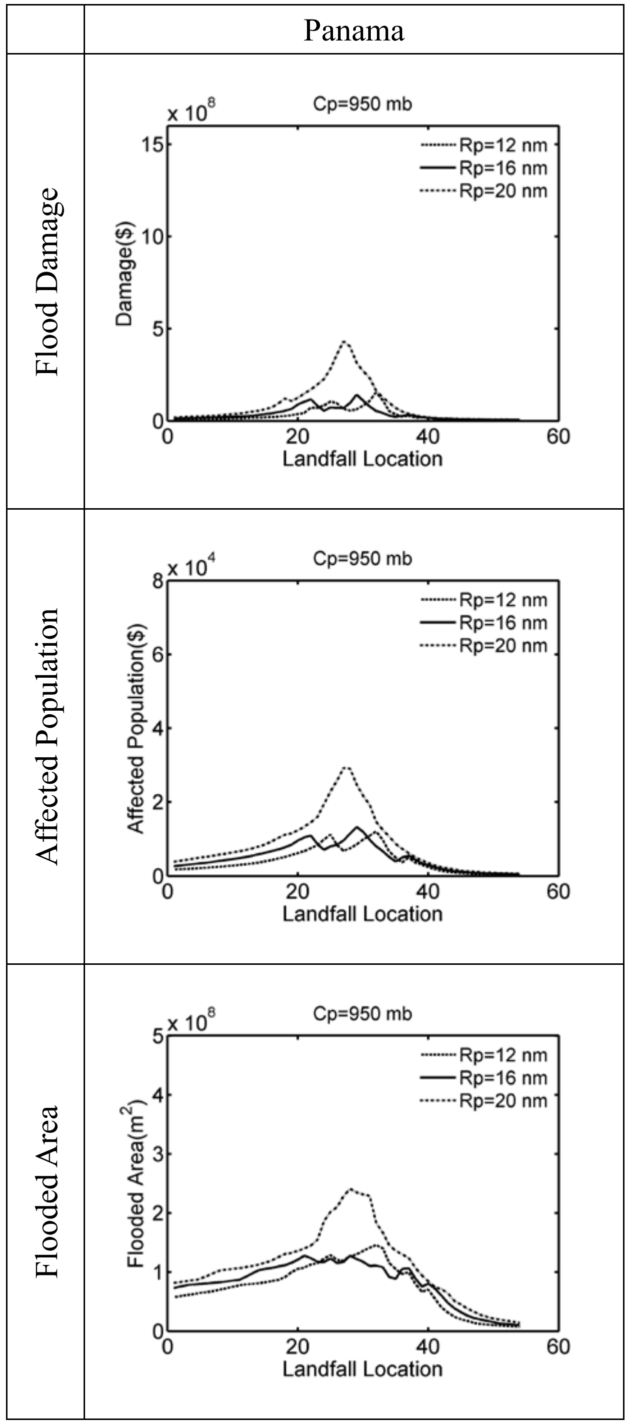
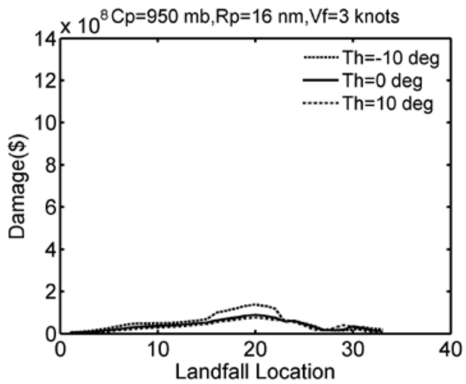
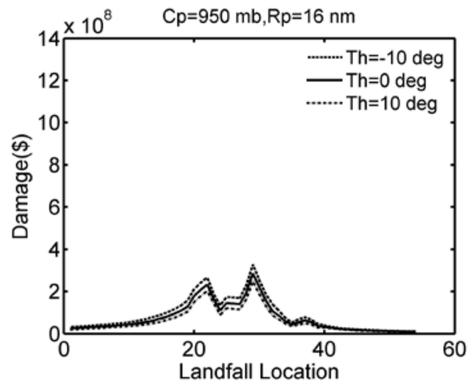


Figure 5-52 The Effect of Radius to the Maximum Wind on Damage Estimation (Panama)

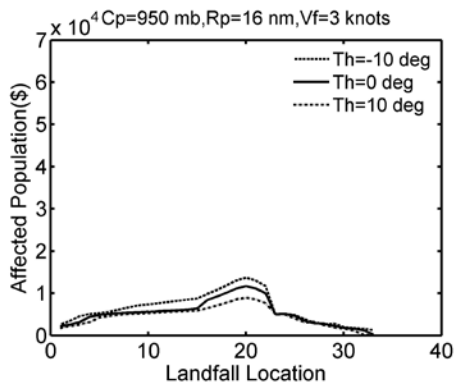
Corpus Christi — Flooded Damage



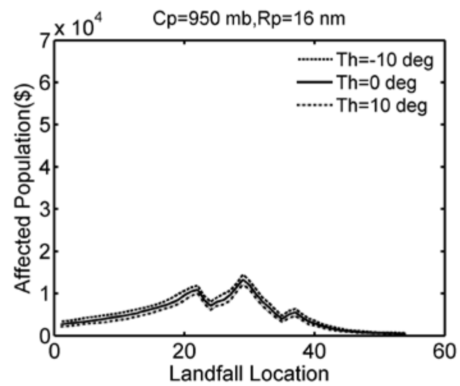
Panama — Flooded Damage



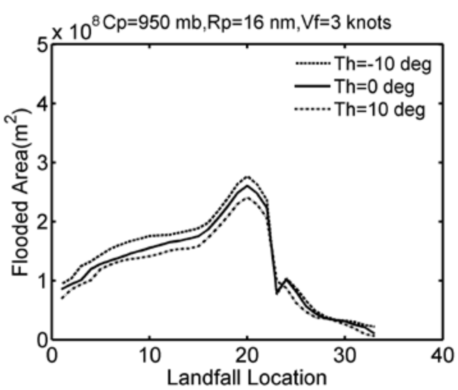
Corpus Christi — Affected Population



Panama — Affected Population



Corpus Christi — Flooded Area



Panama — Flooded Area

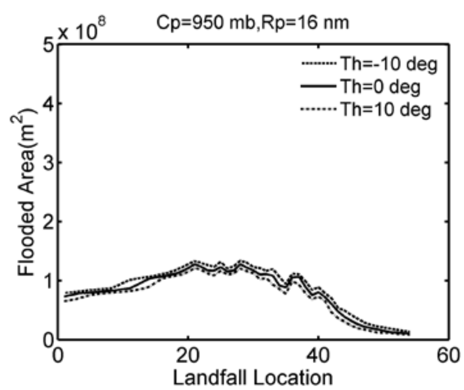


Figure 5-53 The Effect of Approach Angle on Damage Estimation (Corpus Christi and Panama)

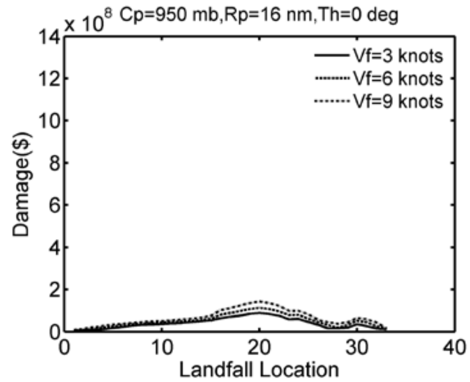
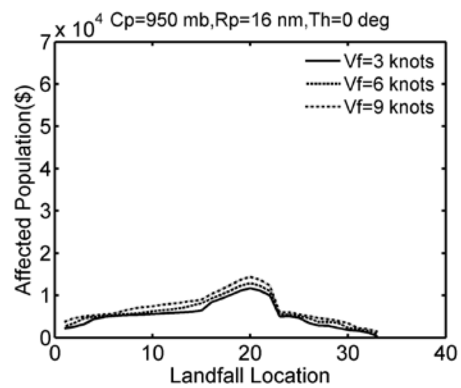
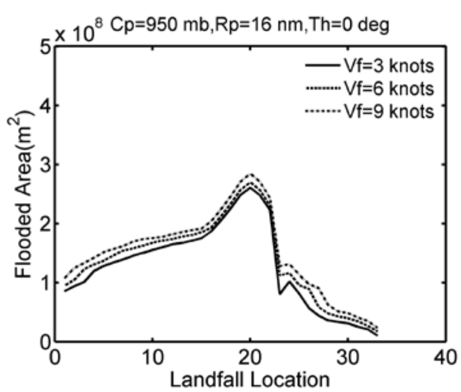
	Corpus Christi
Flood Damage	
Affected Population	
Flooded Area	

Figure 5-54 The Effect of Forward Speed on Damage Estimation (Corpus Christi)

To quantify the error on the estimation of flood damage, affected population and flood area, we calculated the range of error by assuming the surge value are 40 cm greater or less than the value given by the surge response function, as 40 cm is approximately the average root mean square error between surge response function and hydrodynamic model results (Udoh, 2012). Figure 5-55 and Figure 5-56 are the flood damage, affected population and flooded area estimation along with the error range of each value for the hypothetical hurricane (the same as used in section 5.6) making landfall at different locations and assumed current climate condition. It was found that the error on flood damage and affected population are about the same order (approximately ± 100 million dollars of damage, $\pm 5{,}000$ people and ± 40 km^2 flooded area), whereas the range on flooded area in Corpus Christi is greater than that in Gulfport and Panama. This indicates that the flooded area in Corpus Christi is more sensitive to surge values while the damage and affected are not as sensitive as much. Physically it could mean that where the surge floods are mostly the area without properties and unpopulated. The flood damage, affected population and flood area values for hurricane events can be looked together with associated flood maps to check where the places are being flooded. In the following section we develop the flood maps of the worst scenario (when maximum damage is caused) for the three cities.

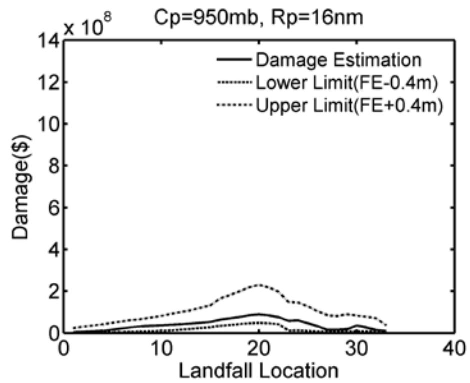
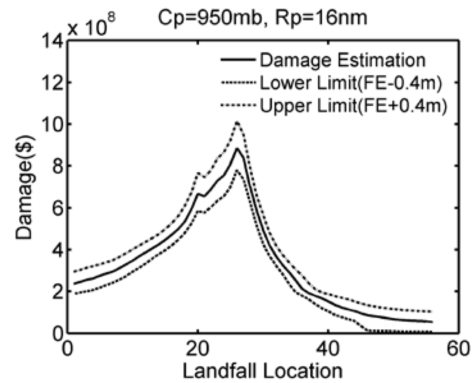
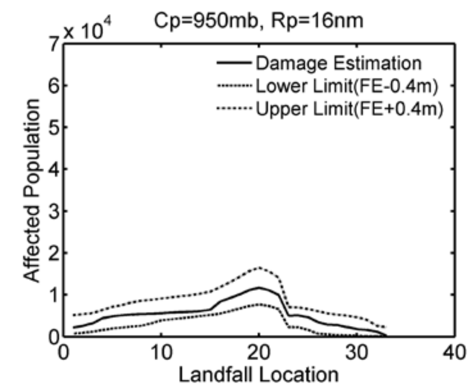
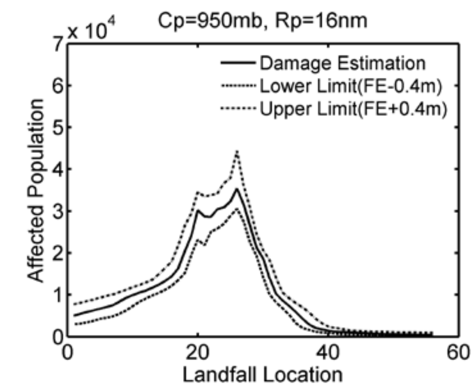
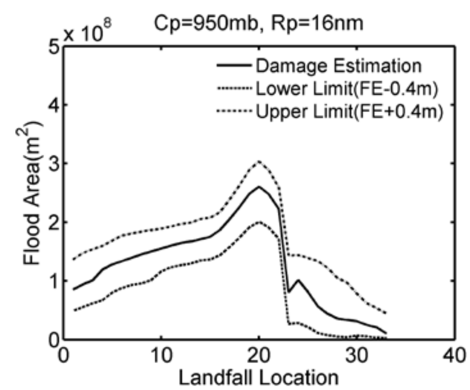
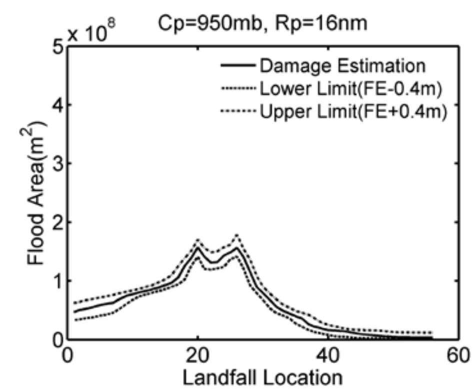
<table>
<tr><th>Corpus Christi, TX</th><th>Gulfport, MS</th></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>
</td><td>
</td></tr>
</table>

Figure 5-55 Sensitivity Analysis of Damage to SRF Surges (Corpus Christi, TX)

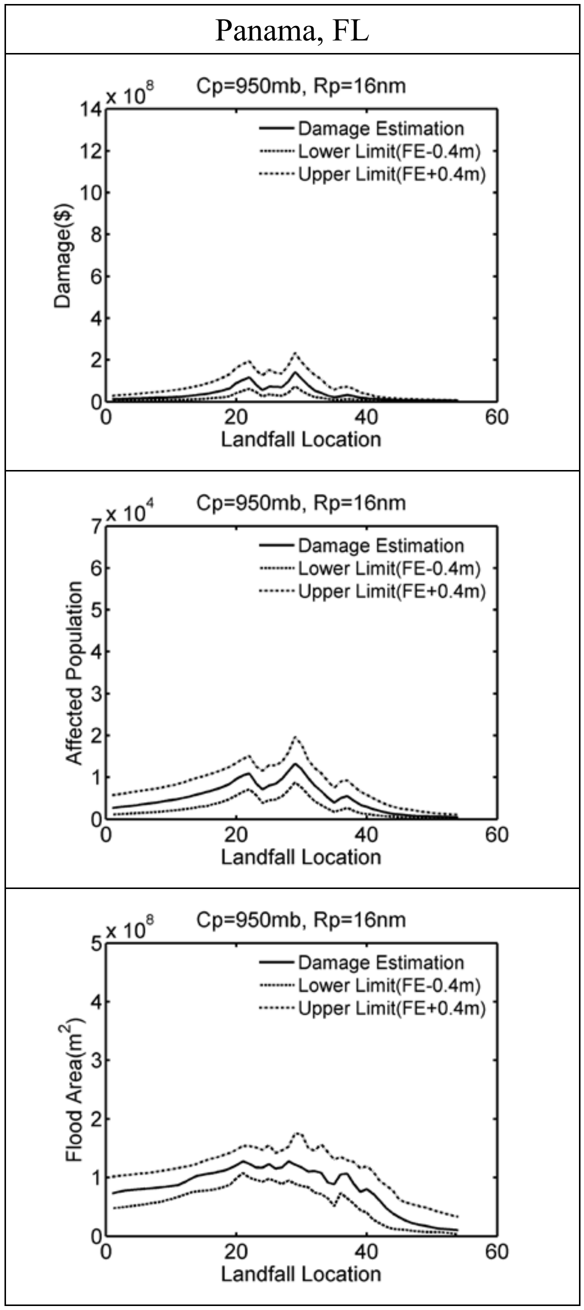


Figure 5-56 Sensitivity Analysis of Damage to SRF Surges (Gulfport, TX)

Table 5–8 through Table 5–10 show the comparison of this study and Frey et al. (2010) results on flood damage, affected population and flooded area estimation at Corpus Christi area. It was found that the damage estimation for current condition and 2030s in this study is less, while the estimation in 2080s is greater than Frey et al. (2010) results. The affected population estimation in this study is about one-third to two-third of the values estimated in Frey et al. (2010). The flooded area estimation in this study is about two to three times of the values estimated in Frey et al. (2010). Except for the damage estimate for current condition, the difference is within order of three.

Table 5–8 Comparison of Flood Damage (in Million Dollars) of Frey et al. and This Study

		Bret(1999)[1] c_p=953 mb R_p=19 km(~10.25 nm)	This Study c_p=950 mb R_p=12 nm
2000s current		7	1
2030s	min(B1)	17	5
	max(A1FI)	28	18
2080s	min(B1)	35	88
	max(A1FI)	280	378

[1]The land fall location of Bret(1999) is approximately   for the Corpus Christi case in this study

Table 5–9 Comparison of Affected Population of Frey et al. and This Study

		Bret(1999) c_p=953 mb R_p=19 km(~10.25 nm)	This Study c_p=950 mb R_p=12 nm
2000s current		5,700	1,600
2030s	min(B1)	6,100	2,300
	max(A1FI)	7,400	4,000
2080s	min(B1)	9,100	5,400
	max(A1FI)	17,100	9,300

Table 5–10 Comparison of Flooded Area (in km^2) of Frey et al. and This Study

	Bret(1999) c_p=953 mb R_p=19 km(~10.25 nm)	This Study c_p=950 mb R_p=12 nm
2000s current	31	65
2030s min(B1)	33	89
max(A1FI)	41	102
2080s min(B1)	50	136
max(A1FI)	101	183

5.7. Flood Maps

Using the flood map developing process discussed in the Methodology section; we have developed flood maps of the hypothetical hurricane with 950 mb of central pressure, 16 nm of radius to maximum wind and makes landfall at location that causes the maximum damage to the three cities (see Figure 5-57, Figure 5-58 and Figure 5-59). The resulting maps show that the flooded area at the three cities match the locations that are subject to high expected annual loss as show in Figure 5-12 through Figure 5-14. From the water depth legend of these maps, we can tell that Gulfport is subject to greater flood depth, which matches the result that Gulfport gets greater flood damage than the other two cities from the strike of the hypothetical hurricane. This map developing process, which incorporate the surge response functions, makes it possible to quickly develop flood maps without running hydrodynamic models. These maps can be used to confirm the results of the damage value estimate and help us to locate properties, population and businesses that are subject to hurricane surge flood risk.

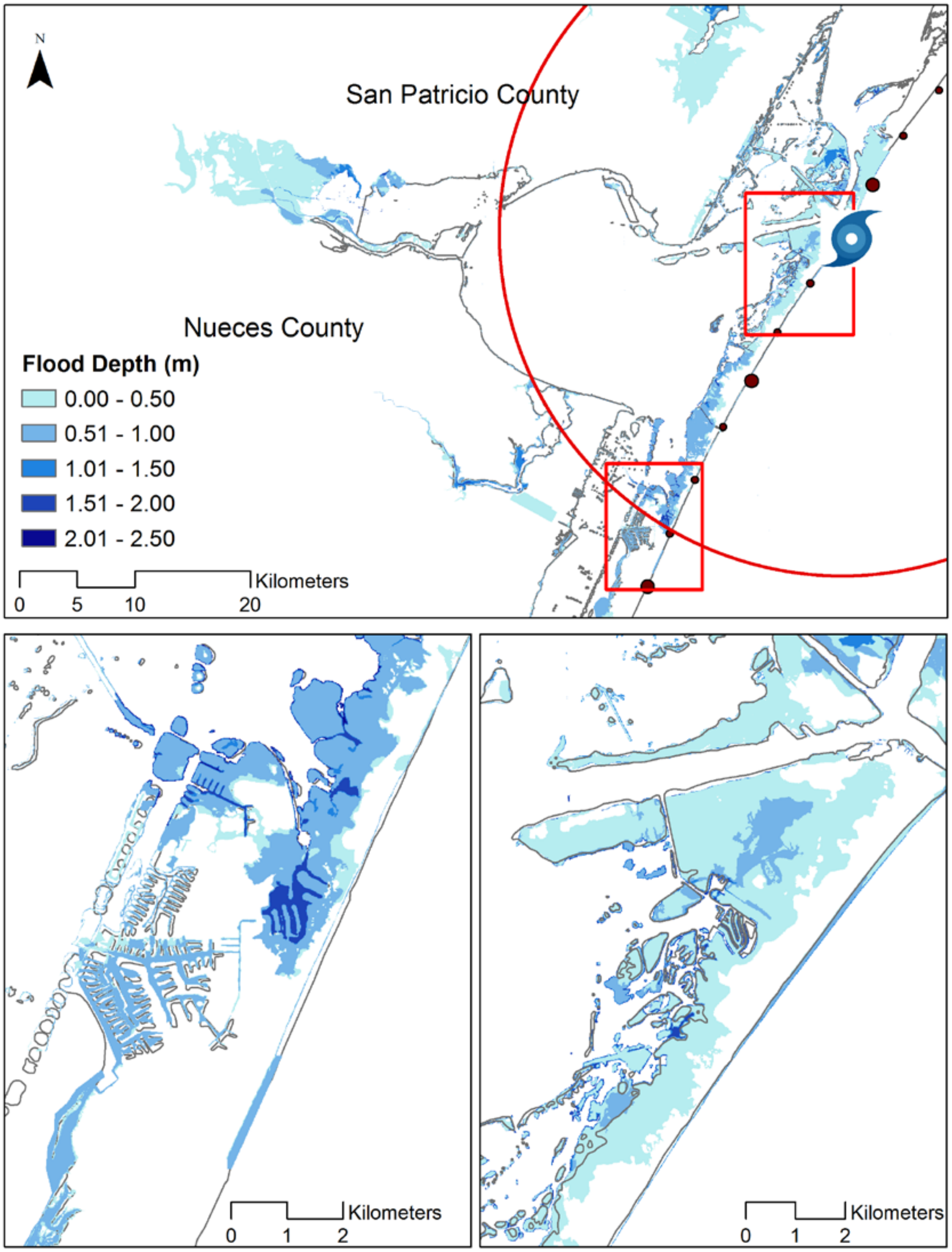


Figure 5-57 Flood Map for Corpus Christi, TX

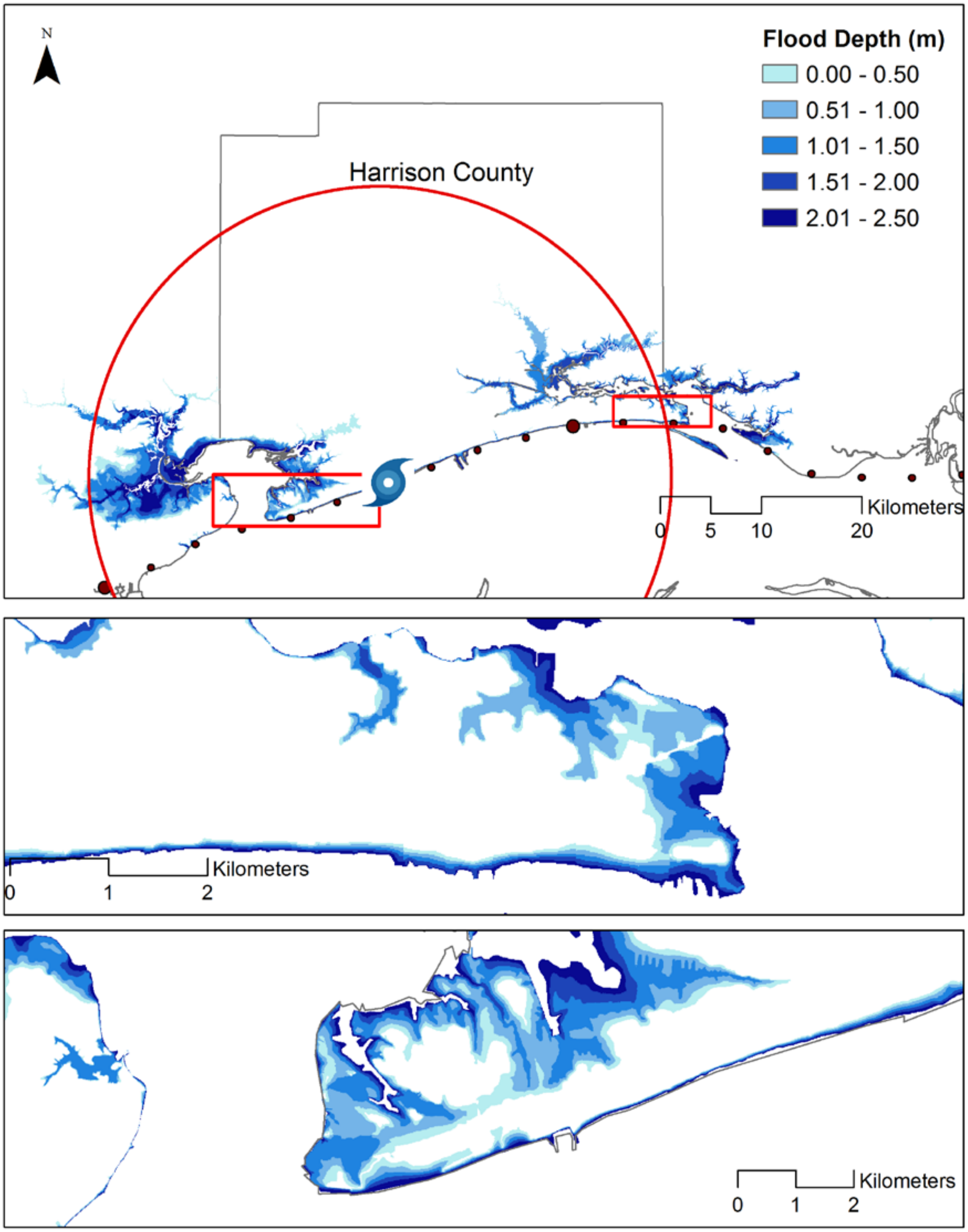


Figure 5-58 Flood Map for Gulfport, MS

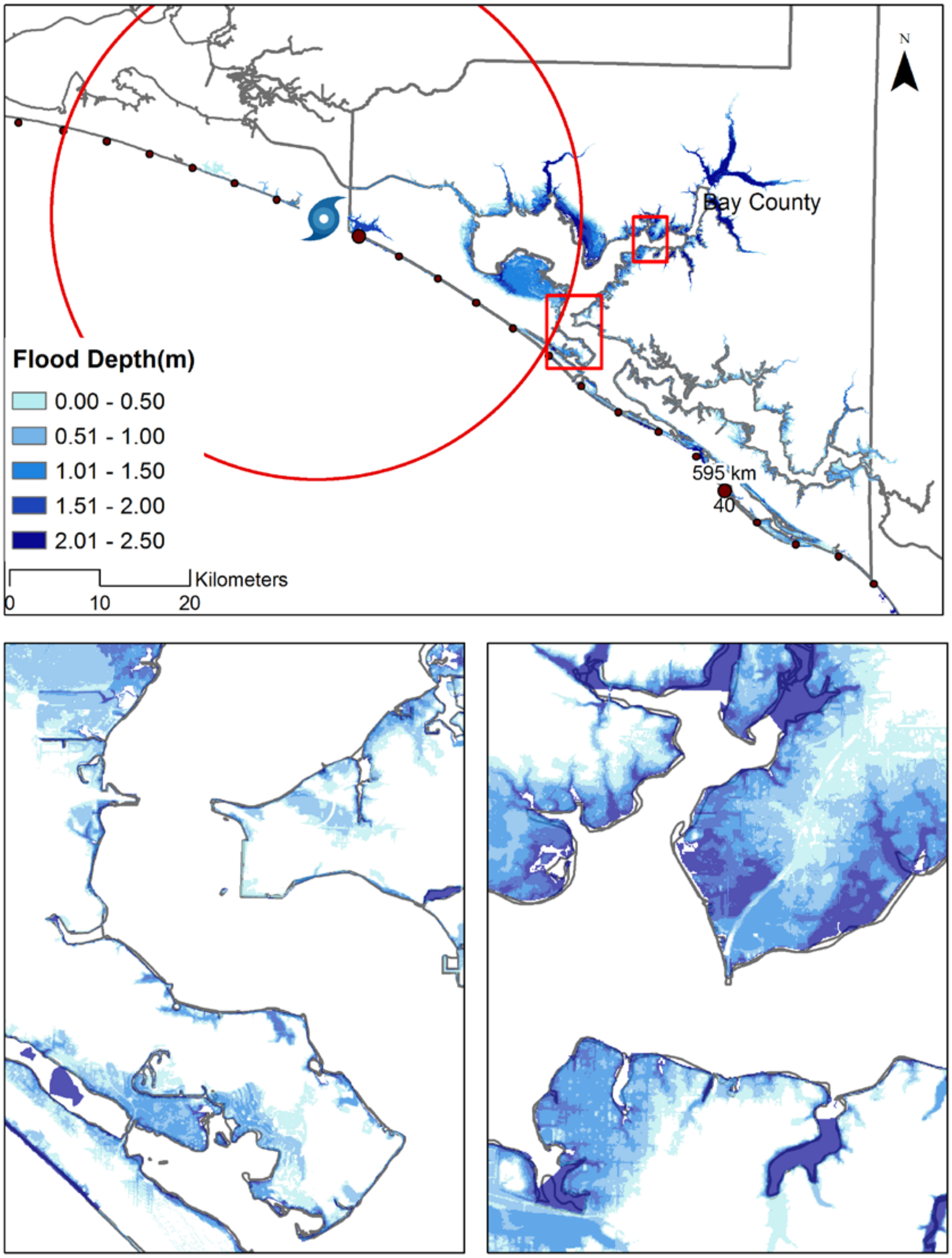


Figure 5-59 Flood Map for Panama, FL

6. WEB-BASED GAME DEVELOPMENT TO SUPPORT K12 EDUCATION FOR UNDERSTANDING CLIMATE CHANGE IMPACT ON HURRICANE FLOODING DAMAGE

Educating the public on the effects and risk of hurricanes is as important as studying the hurricane effects and risk themselves. Game-based tools have been used for education and training for more than a decade (Luo et al., 2008). Educational games with interactive environments can draw the attention of students and also have fundamentally sound learning principles (Gee, 2004). University Corporation for Atmosphere Research developed an interactive online course “Hurricane Strike!” that teaches the public hurricane preparedness (UCAR, 2012). “Hurricane Strike!” provides a story line that the user can go through and make decisions throughout a virtual hurricane strike. The instructions will lead the user to make proper decisions. National Geographic’s website “The Eye of the Hurricane” (Natioal Geographic, 2012) provides a series of material such as Flash movies and quizzes, which introduces the structure of a hurricane eye. (Luo et al., 2008) developed “Hurricane!” that teaches students about the effects of hurricane winds on different kinds of residential structures. This program is developed based on a graphics engine and a physics engine, which provide a physics-based animation for to the users.

Aside from the above-mentioned educational tools, there are a number of materials online that provide text, image and animations that teach the public about the formation of hurricanes and their effects. However, few of them were developed to tell the effects of climate change on hurricanes. Here we applied the concept of "tabular approach" to develop a web-based game "VisHurricane", as the damage value is pre-calculated and stored in a database, and retrieved when given hurricane parameters. This game is intended to give K-12 students a sense of how climate change could make difference on our future. "VisHurricane!" allow students to choose different future climate scenarios and see how climate change could impact the results of hurricane flooding. Through the game the students can also make decisions to reduce the damage from hurricane events. The supplemental references listed on the website also provide the students the information regarding climate change and hurricane induced damage.

User's Manual for "VisHurricane" is provided for users' reference (see Appendix I). Appendix J also provides the instruction of website server configuration, which allows people to install "VisHurricane" on any personal computer with Windows 8 operation system and connected to the internet.

6.1. The Mechanism of VisHurricane

VisHurricane considers three virtual cities along the coastline, and annual simulated virtual hurricane events flood areas, cause property damage, and affect population (people that live in the flooded areas). There is a fixed probability for a hurricane to develop in a simulation year. Once a hurricane is developed, it could strike one of the three cities, or a place without population and caused no damage. VisHurricane runs from the current year until year 2100. Before VisHurricane simulates a hurricane event, the user has to choose one of the IPCC (Intergovernmental Panel on Climate Change) scenarios which give different projected of CO^2 concentration (Figure 6-1) and temperature changes up to year 2100 (Figure 6-2).

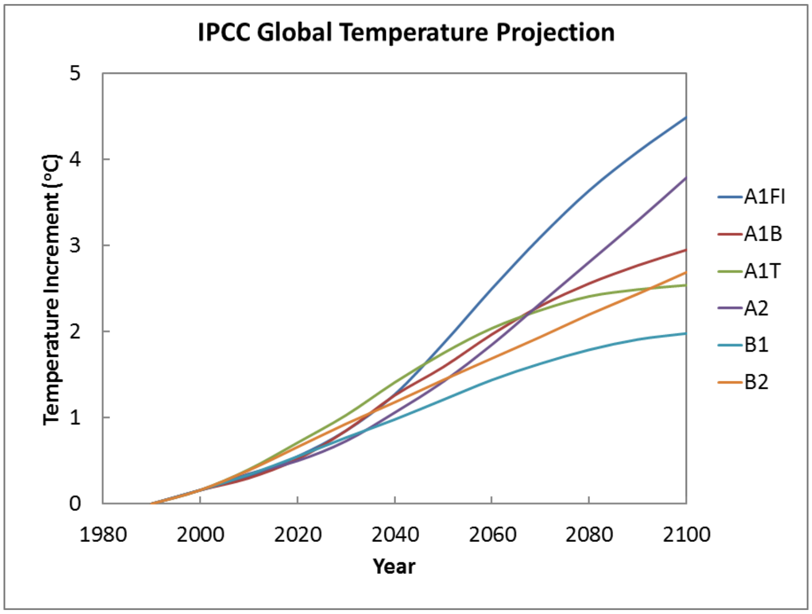


Figure 6-1IPCC Global Temperature Projection (AR3)

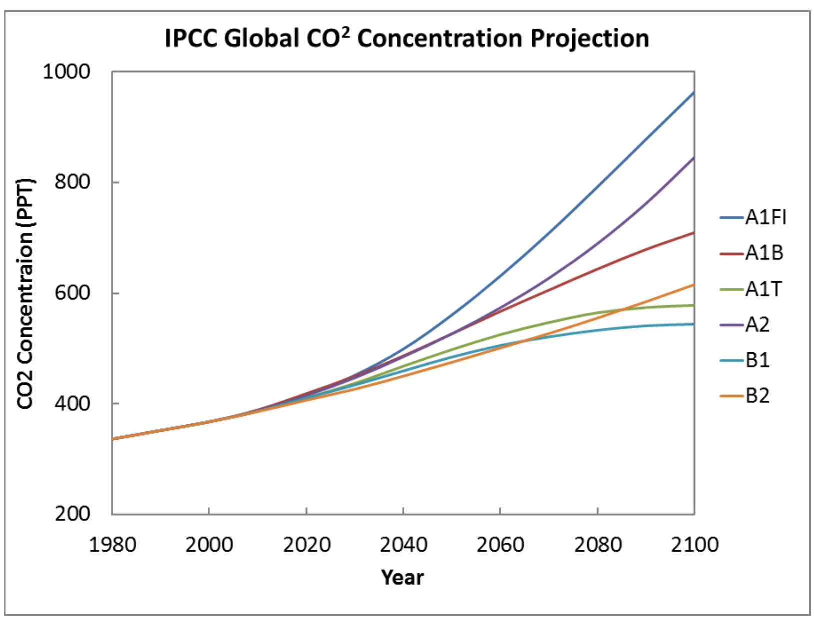


Figure 6-2 IPCC Global CO^2 Concentration Projection (AR4)

According to Knutson and Tuleya (2004), the hurricane intensity (or the central pressure) will change with the sea surface temperature change, resulting in different amounts of damage. In VisHurricane, we assumed that the sea surface temperature change is the same as the surface air temperature change and uses the values in the IPCC Third Assessment Report (GRID-Arendal, 2013) for simulation.

The user can choose the frequency in which input is entered. For example, if the user chooses ten years, the simulation will stop every ten years and ask for input. Every time the user proceeds to the next period, he/she can change the budget assigned to each city for damage mitigation. Before proceeding to the next simulation period, the user will see the statistics (the summation) of property damage, affected population and flooded area of the three cities in previous period (except for the first one). The user can use the statistics for making decisions (changing the allocation of budget) for next period. At the end of the simulation, a panel will display the damage records (the property damage, affected population, and flooded area) for all the simulation years and compare the damage occurred at each city. By playing the game many times, the user will see the different results based on different climate scenarios.

The flowchart of VisHurricane is shown as Figure 6-3.

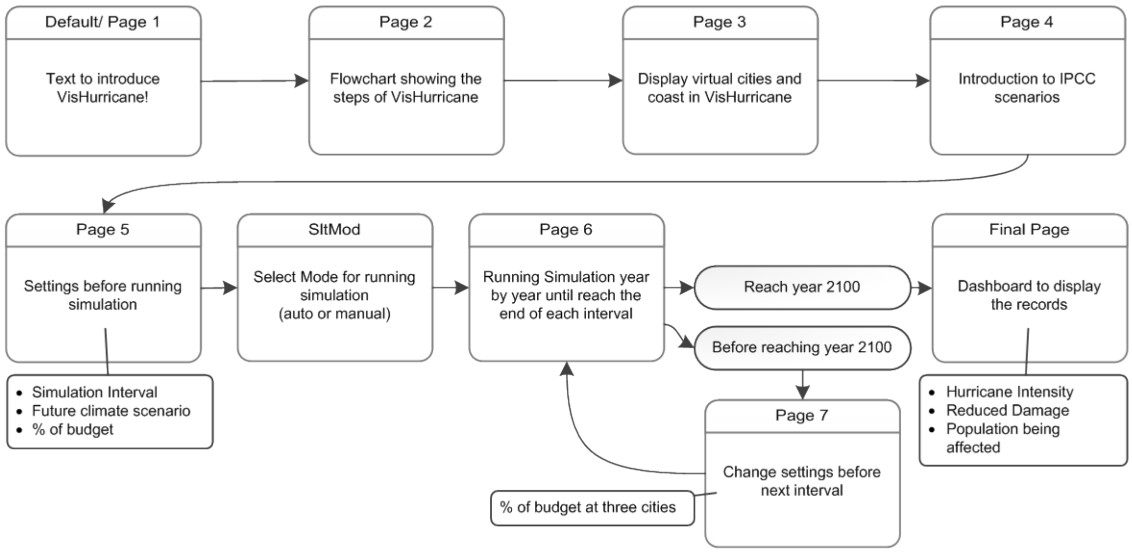


Figure 6-3 VisHurricane Flowchart

6.2. Data Needed for VisHurricane

Three types of information are needed for VisHurricane: Hurricane Names, which is used to displayed on the screen whenever a hurricane event occurs; the IPCC projected temperature and CO^2 concentration for different climate scenarios, which is used as the input of the embedded program, in order to simulate the hurricane event.; and a lookup table of the flood damage, affected population and flooded area for different values of hurricane intensity (central pressure) and hurricane size (radius to the maximum wind), which is used to lookup the damage value after the program simulates the hurricane intensity and size. The Hurricane Name is obtained from the “Atlantic Tropical Cyclone Names” on National Hurricane Center website (NOAA, 2013). The CO^2 concentration for different climate scenarios are obtained from IPCC Data Distribution Centre (IPCC, 2013). The flood damage, affected population and flooded area according to hurricane intensity (central pressure) and hurricane size (radius to the maximum wind) are hypothetical values consistent with the damage analysis of Corpus Christi, Gulfport and Panama City.

All the information is stored in an App_Data folder. The hurricane names are stored in a HurricaneNames.mdb file. The projected CO^2 concentration and temperature change is stored in the IPCCprj.mdb file. The damage, affected population, flooded area according to hurricane intensity (central pressure) and hurricane size (radius to the maximum wind) is stored in the HurrMx.mdb file. Inside HurrMx.mdb, there are three data tables, one for each city within VisHurricane. In each data table, five fields are specified: the first one is the hurricane central pressure; the second field is the hurricane radius to the maximum wind; the third field is the property damage value; the fourth field is the affected population; and the fifth one is the flooded area.

6.3. Purpose and Code Functions of Each Page in VisHurricane

6.3.1. Front page (Default.aspx)

The default page is the portal page of VisHurricane, which gives the brief introduction of the game. The button at the bottom center redirects to Page02. (See Appendix H-H.1 for the page code.)

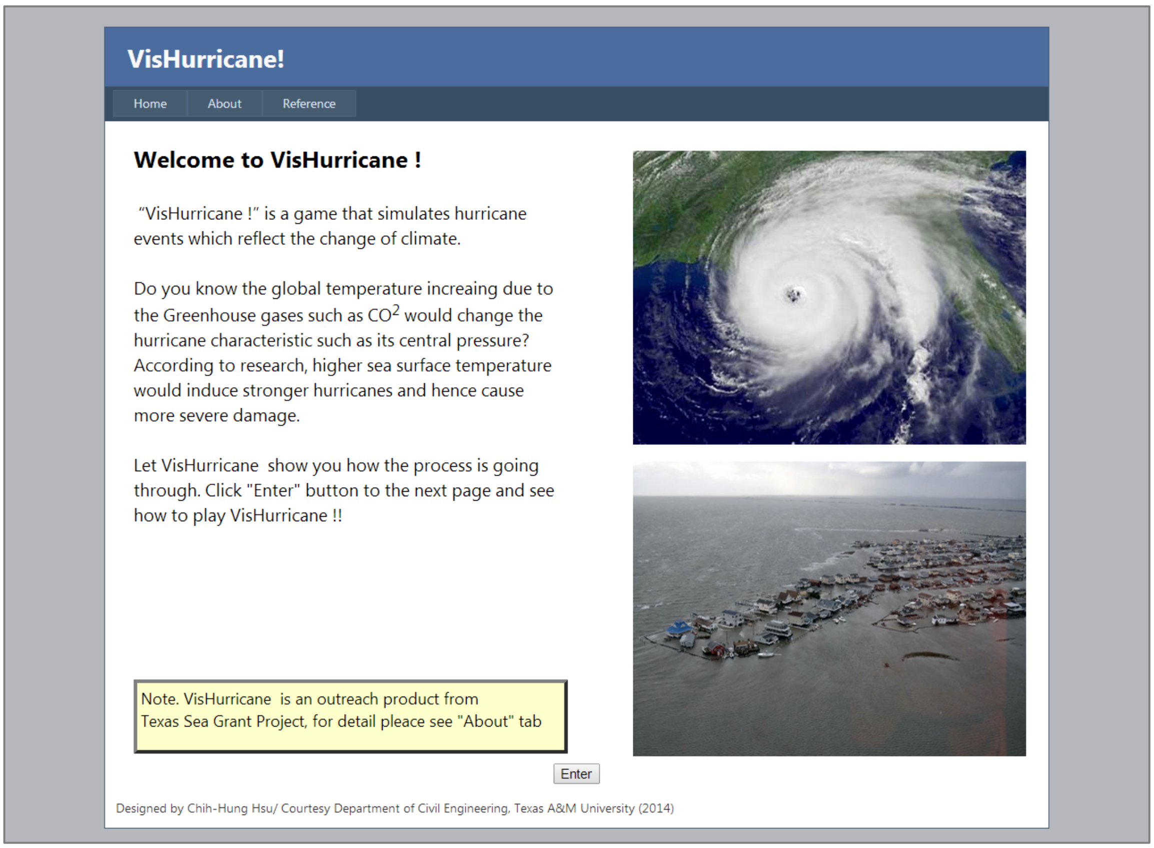


Figure 6-4 Default.aspx

6.3.2. Page02

Page2 shows the steps of playing VisHurricane. The button at bottom center redirects to Page03. (See Appendix H-H.2 for the page code.).

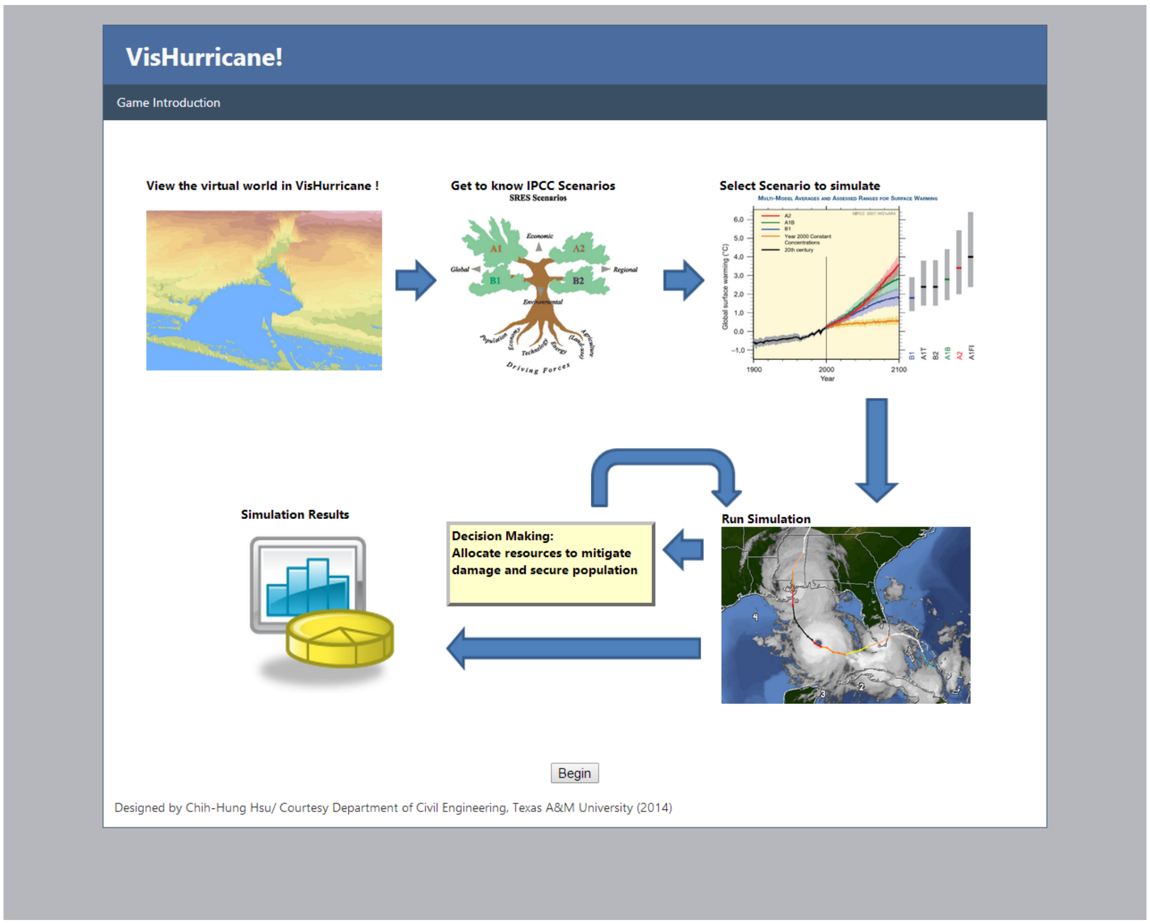


Figure 6-5 Page02.aspx

6.3.3. Page03

Page3 displays the virtual cities and coastline. The radio button at the bottom left allows the user to select a city. The image at bottom center will change according to user's selection. The text at bottom right will change according to user's selection as well. (See Appendix H-H.3 for the page code)

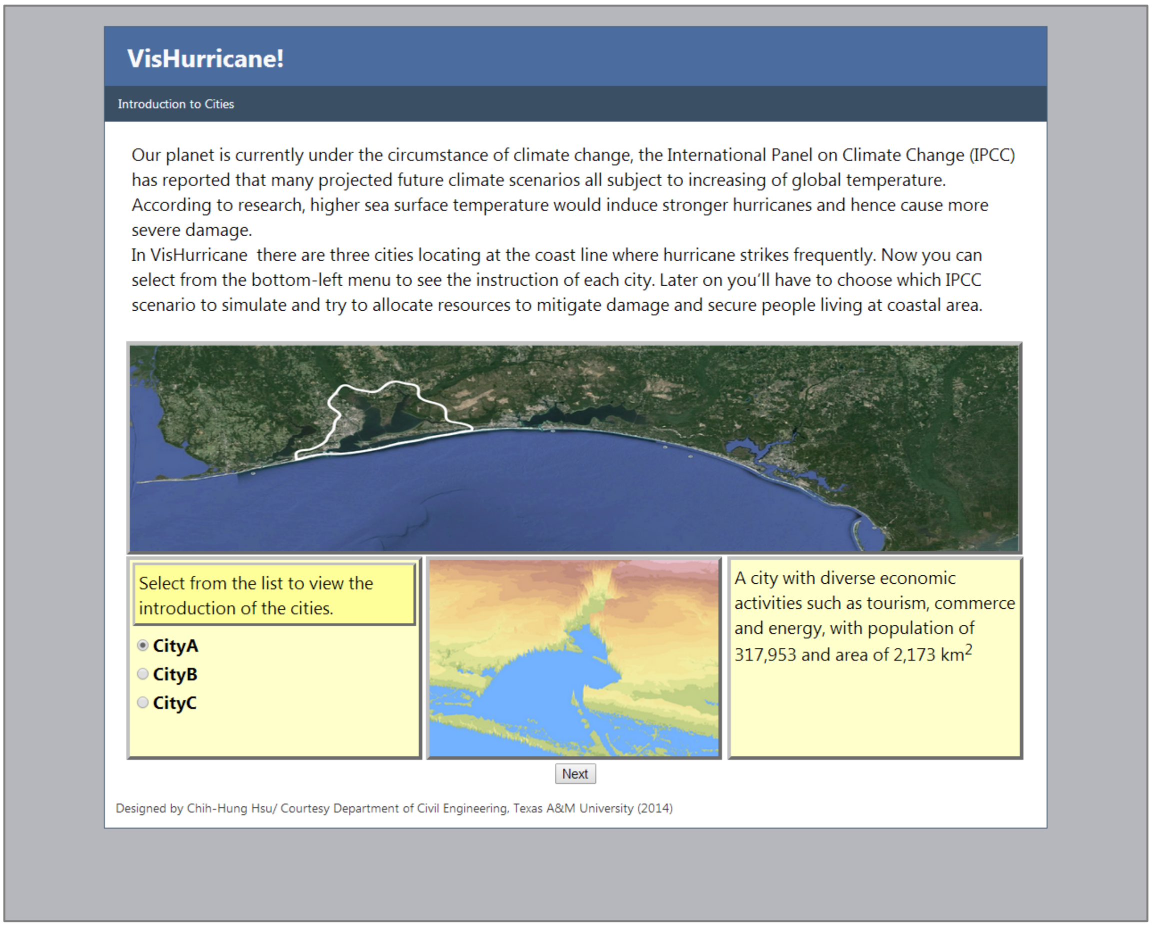


Figure 6-6 Page03.aspx

6.3.4. Page04

Page4 gives the introduction to IPCC future climate scenarios. The radio button list at the center left is designed for the user to select a IPCC scenario (IPCC, 2007a). The image at the center right and bottom right and the text at bottom left will display the CO^2 concentration and temperature increment based on the selected future climate scenarios; the button at bottom center is to redirect the page to Page05. (See Appendix H-H.4 for the page code.)

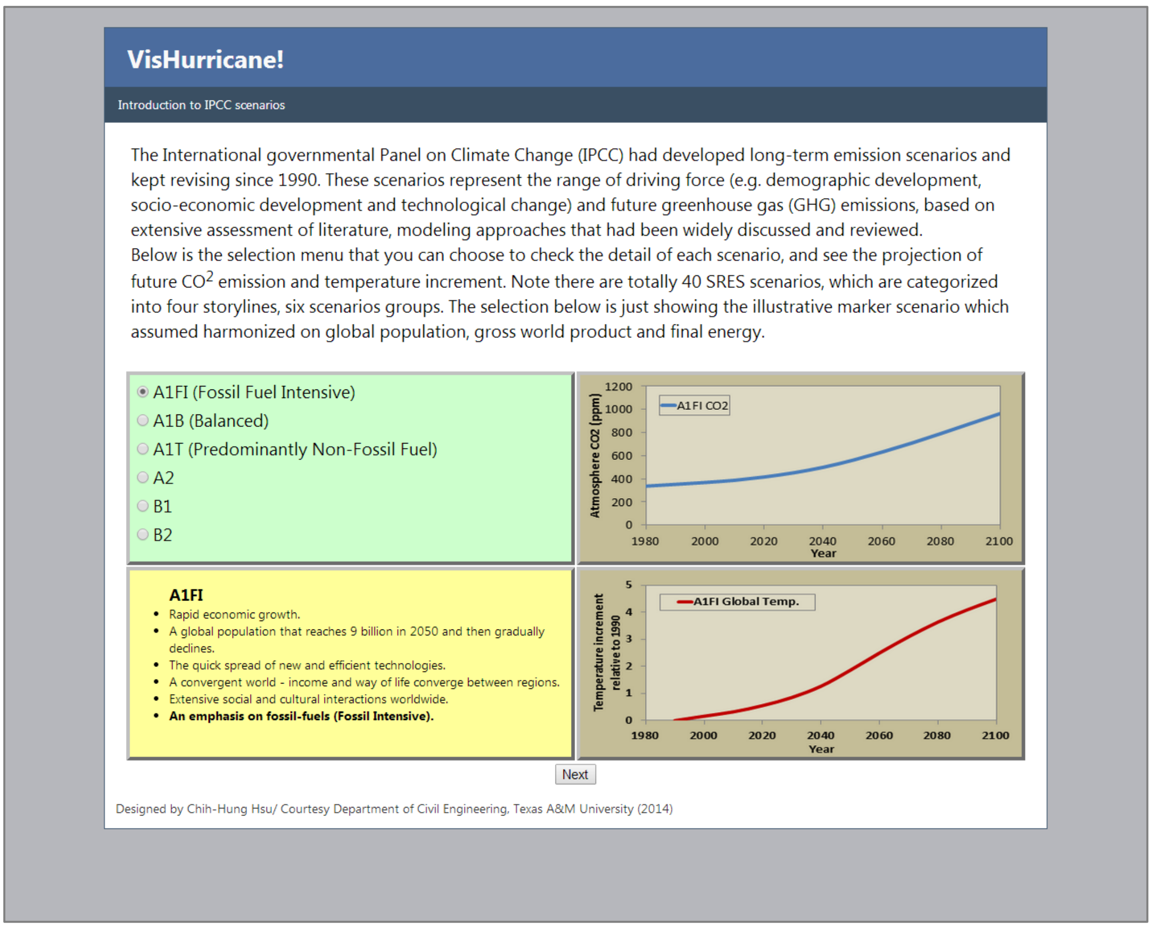


Figure 6-7 Page04.aspx

6.3.5. Page05

Page5 is for the user to specify the simulation period (frequency with which the user can change the settings), IPCC scenario (defines the CO^2 concentration and temperature change projection) and the resources allocation (defines the amount of budge to spend on each of the three cities.) The radio button list on the top center is for selecting the simulation period; the radio button list at middle center is for selecting the IPCC scenario; the dropdown list at bottom center is for allocating the budget, which is the budget that the user can use per year. The annual amount of budget is randomly assigned as 80, 90 or 100 million dollars. For each simulation period, the annual budget will be the same. Every time when reaching the end of each period, the annual budget will be reassigned again .Upon selecting the IPCC scenario, the temperature and CO^2 values for the selected scenario will be retrieved from IPCCprj.mdb. The button at bottom center is to redirect the page to the Selection Mode page. (See Appendix H-H.5 for the page code)

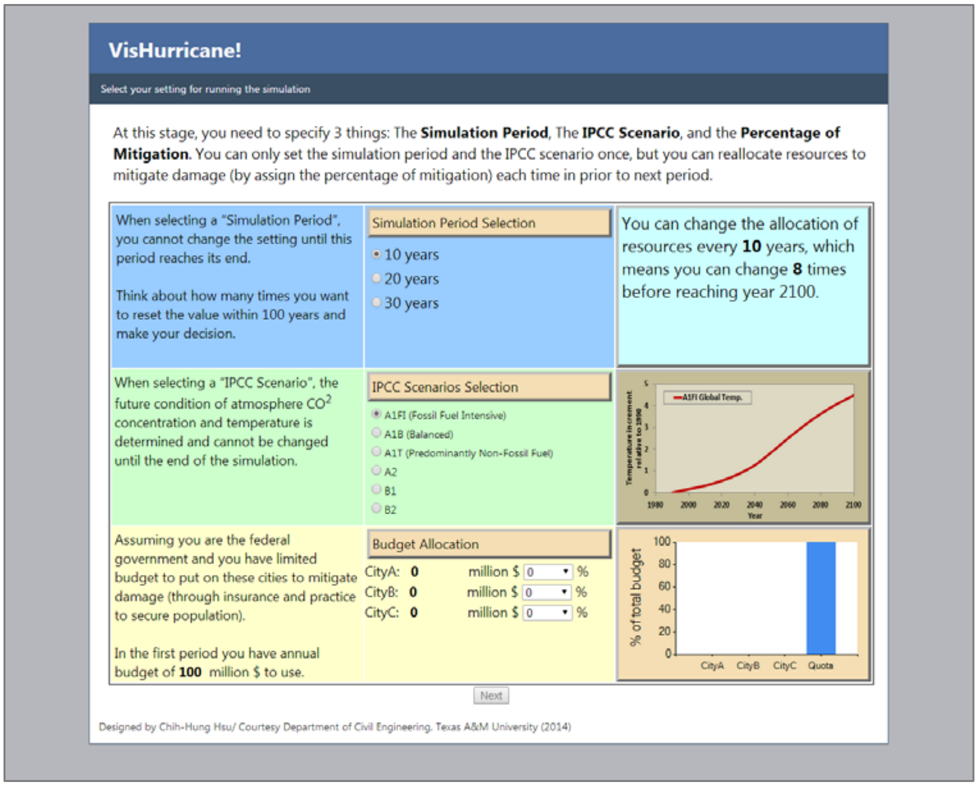


Figure 6-8 Page05.aspx

6.3.6. Mode Selection Page

This page allows the user to select whether to let the computer run the simulation automatically or manually. In Manual mode, the user has to click after each year of simulation and enter input at the end of the period. In Auto Run mode, the user only has to enter input at the end of the period. (See Appendix H-H.6 for the page code)

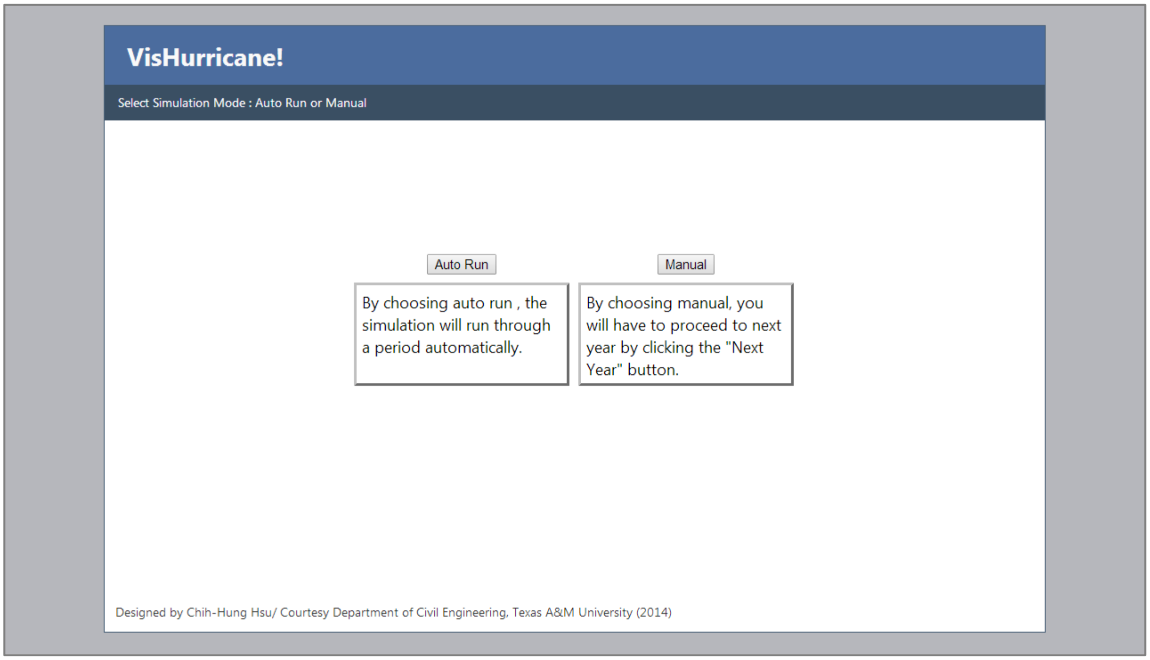


Figure 6-9 ModSlt.aspx

6.3.7. Page06

Page 6 is the simulation page. The image at the top displays an animation of a hurricane striking the coastline. On the center left, information such as current CO^2 concentration and global temperature is displayed. On the bottom left, the hurricane name and category are displayed. On the bottom right, the damage caused by the hurricane, the population affected and the flooded area are displayed.

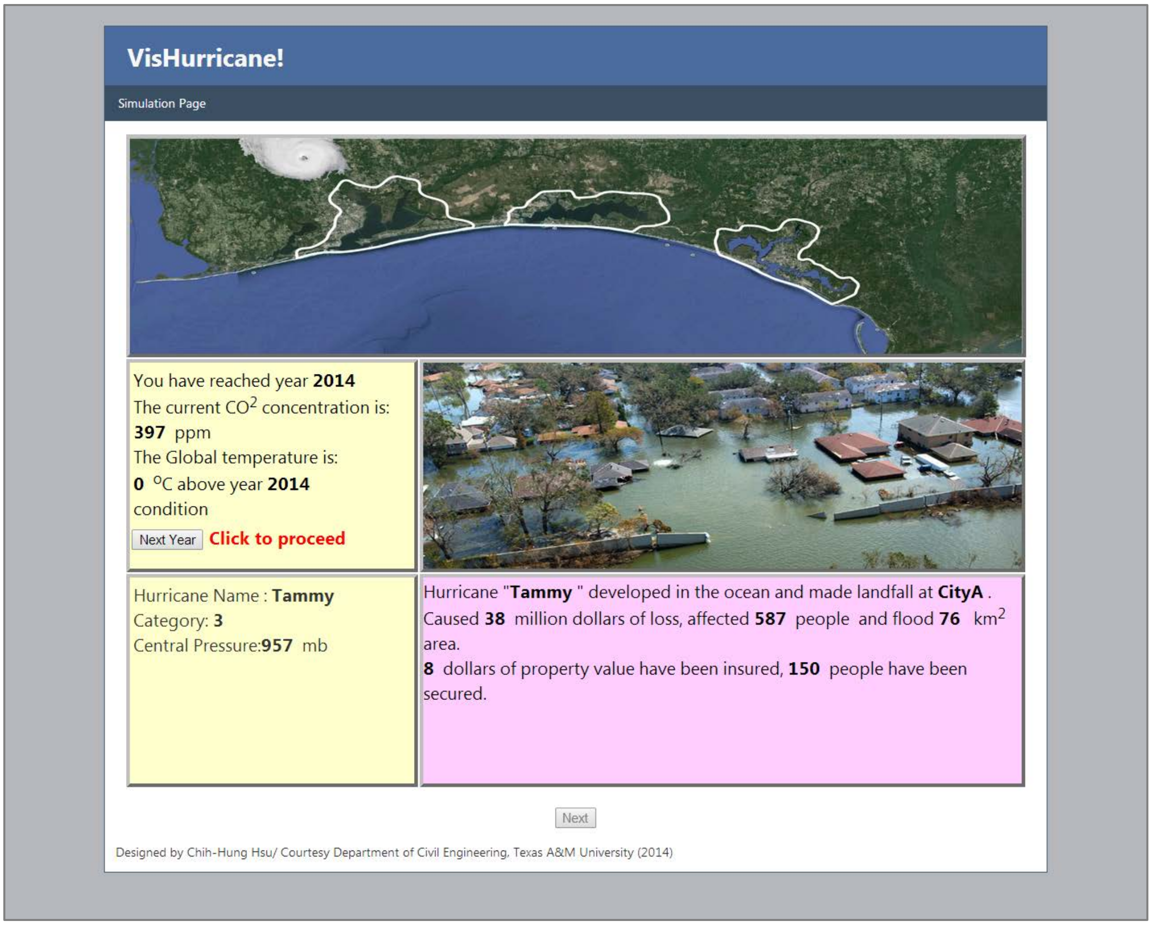


Figure 6-10 Page06.aspx

Table 6-1 shows the list of functions written in C# for running the simulations. In every simulation year, the function Sim() is called to simulate hurricane events and gives the hurricane name, category, radius, central pressure, landfall location, damage, affected population and flooded area.

Figure 6-12 shows the flowchart of the code. (See Appendix H-H.7 for the page code)

Table 6-1 Function Names and Mechanics for Page06.aspx

Function Name	**Function Mechanics**
Sim()	Main function, calls the other functions to simulate hurricane events
bool getOcc()	Simulates the occurrence of hurricane. If returns TRUE, hurricane occurs; if returns FALSE, no hurricane occurs. In the demo version of VisHurricane, the probability of occurrence is set to 50%.
int getLFl()	Simulates the landfall location. If returns 0, hurricane makes landfall at west of City A. If returns 1, hurricane makes landfall at City A. If returns 2, hurricane makes landfall at City B. If returns 3, hurricane makes landfall at City C. If returns 4, hurricane makes landfall at east of City C. (See Figure 6-11 for landfall location specification)
double get CpT(double)	Simulates the hurricane central pressure (C_pT) given the sea surface temperature at each simulation year. C_pT decreases based on the following equation. $C_pT = C_pT_0 - 0.08 \times \Delta SST \times \left(p_{far} - C_pT_0\right)$ where C_pT_0 is the initial C_pT ΔSST is the sea surface temperature increment p_{far} is the far-field barometric pressure In the demo version, C_pT_0 is set to be 945mb, p_{far} is set to be 1013mb
double getCpR(double)	Simulates the actual (real) hurricane central pressure (C_pR) given the theoretical values. In the demo version, the distribution of C_pR is set to be a continuous uniform distribution that spans 50 mb and centered at C_pT.
int getHurCat(double)	Simulates the hurricane category given the actual hurricane central pressure (C_pR).

Function Name	Function Mechanics
	If $C_pR \geq 980mb$ Category =1. If $980mb > C_pR \geq 965mb$ Category =2. If $965mb > C_pR \geq 945mb$ Category =3. If $945mb > C_pR \geq 920mb$ Category =4. If $C_pR < 920mb$, Category =5.
double getRp()	Simulates the hurricane radius (R_p). The distribution of R_p is a discrete uniform distribution that ranges from 4 nm to 64 nm with increment of 4nm.
int getSnoLn()	Gets the line number in the lookup table (HurrMx.mdb) given the actual central pressure (C_pR) and hurricane radius (R_p). With the line number, the value of damage, affected population and flooded area can be obtained from the lookup table (HurrMx.mdb).

Figure 6-11 Illustration of Hurricane Landfall Locations

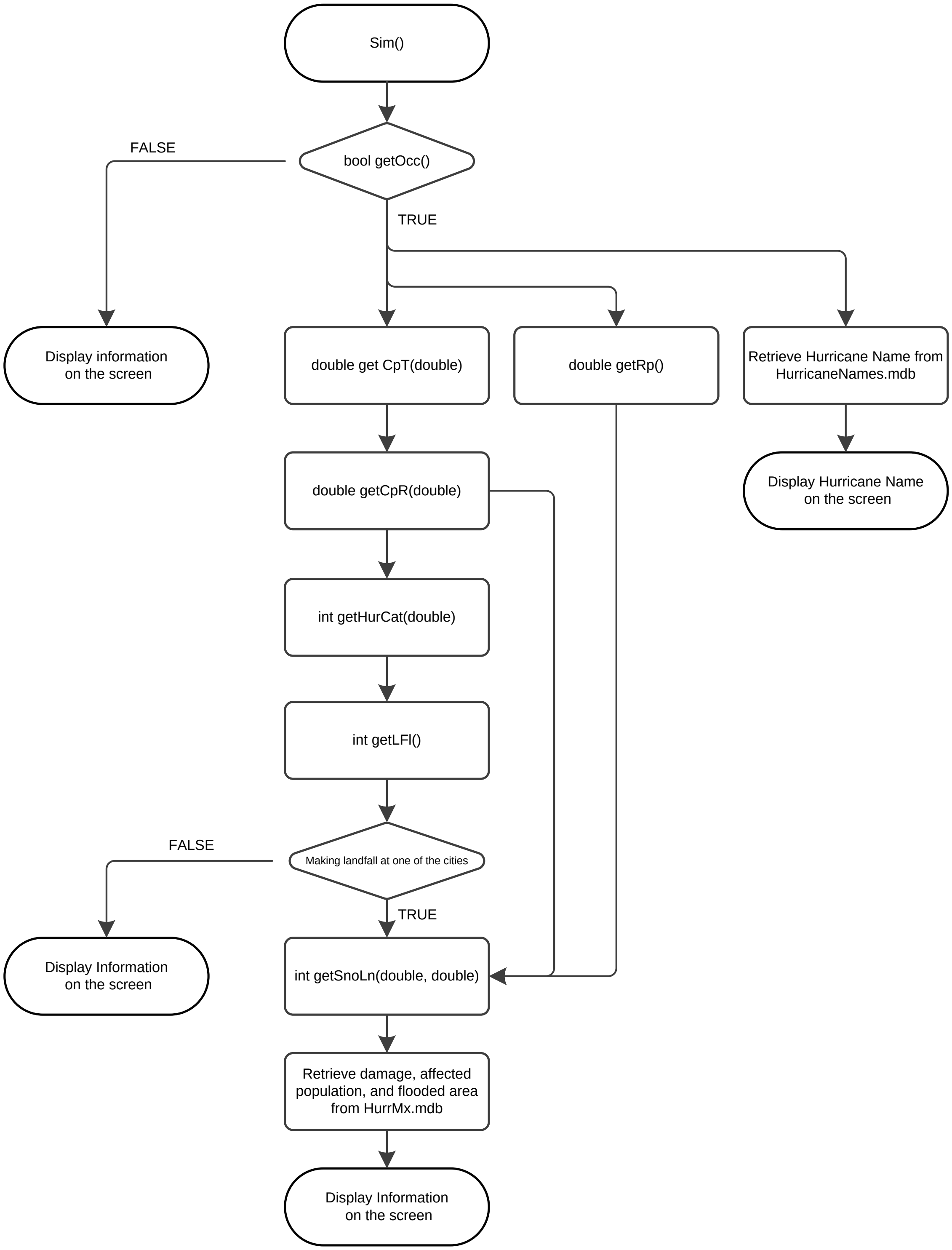


Figure 6-12 Flow Chart of Sim()

6.3.8. Page07

Page7 is the settings page for the next period. In this page, the user can see the statistics of the previous period and can reallocate the budget before running next period. The statistics correspond to the property damage, affected population and flooded area for the simulation period. The property damage and affected population are mitigated in proportion to the budget assigned to each city. (See Appendix H-H.8 for the page code)

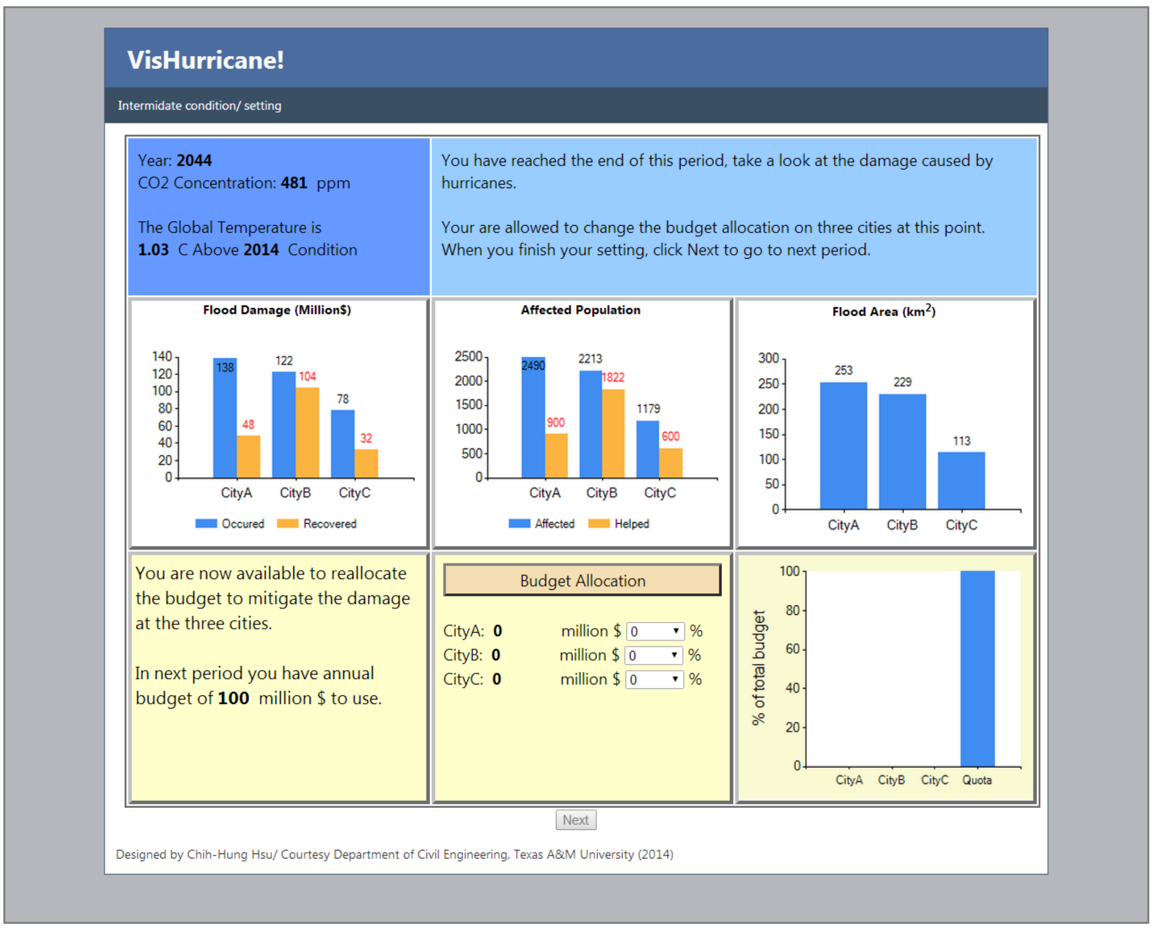


Figure 6-13 Page07.aspx

6.3.9. Final Page

The final page (Figure 6-14 through Figure 6-16) shows the statistics of property damage, affected population, flooded area, and change in hurricane central pressure of all the simulation years (from the beginning to year 2100). (See Appendix H-H.9 for the page code)

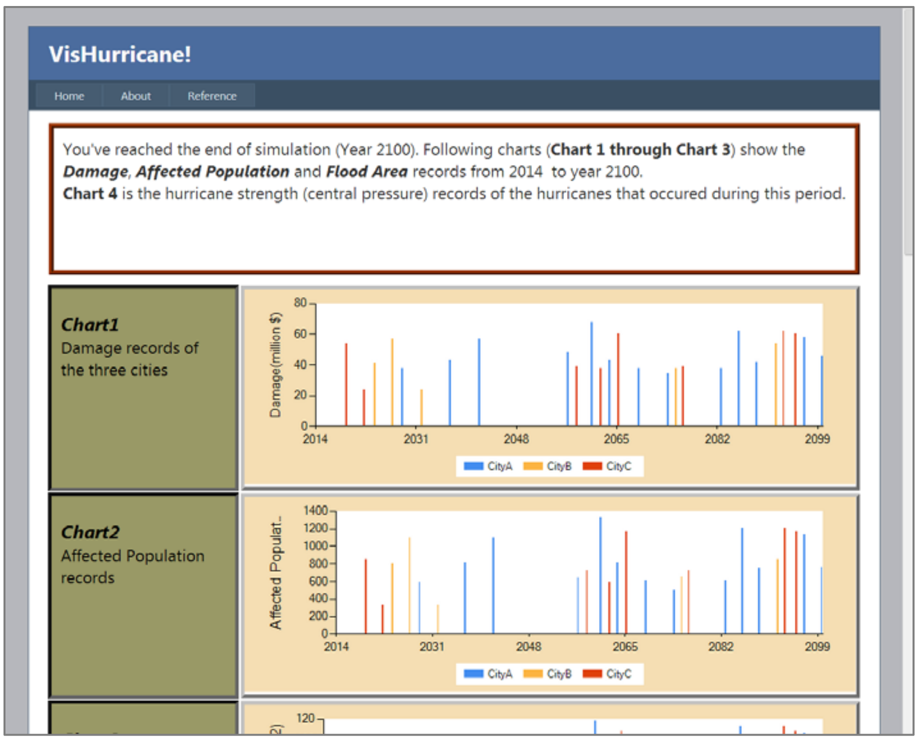


Figure 6-14 Final.aspx(1)

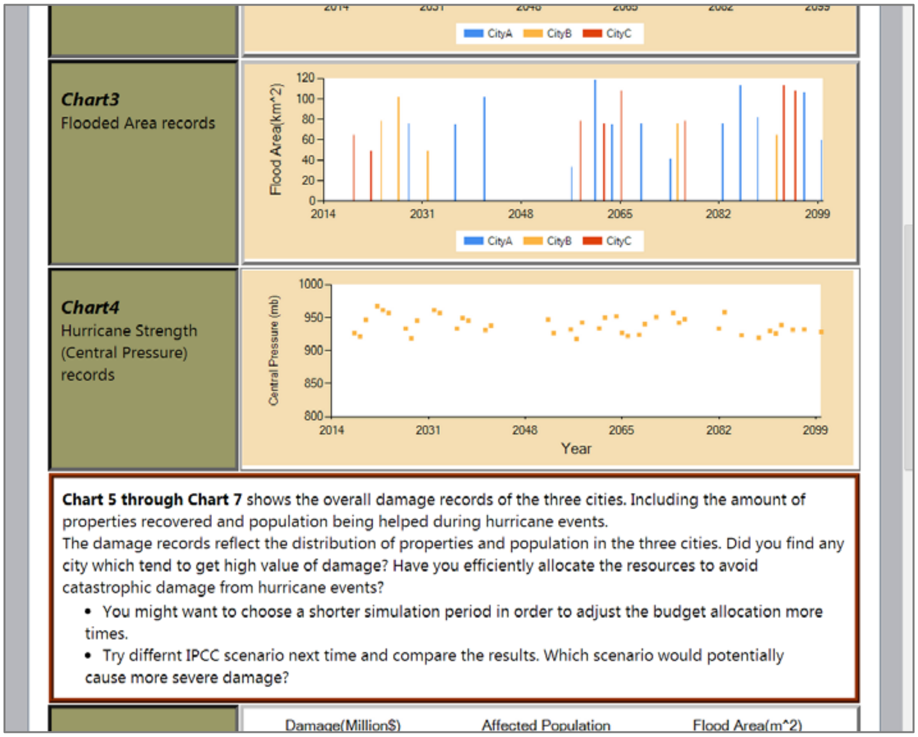


Figure 6-15 Final.aspx(2)

Figure 6-16 Final.aspx(3)

6.3.10. About Page

The page shows the information about SeaGrant project and the purpose of creating VisHurricane.

Figure 6-17 About.aspx

6.3.11. Reference Page

Reference page (Figure 6-18 and Figure 6-19) gives references about climate change, CO^2 emission and hurricane hazard.

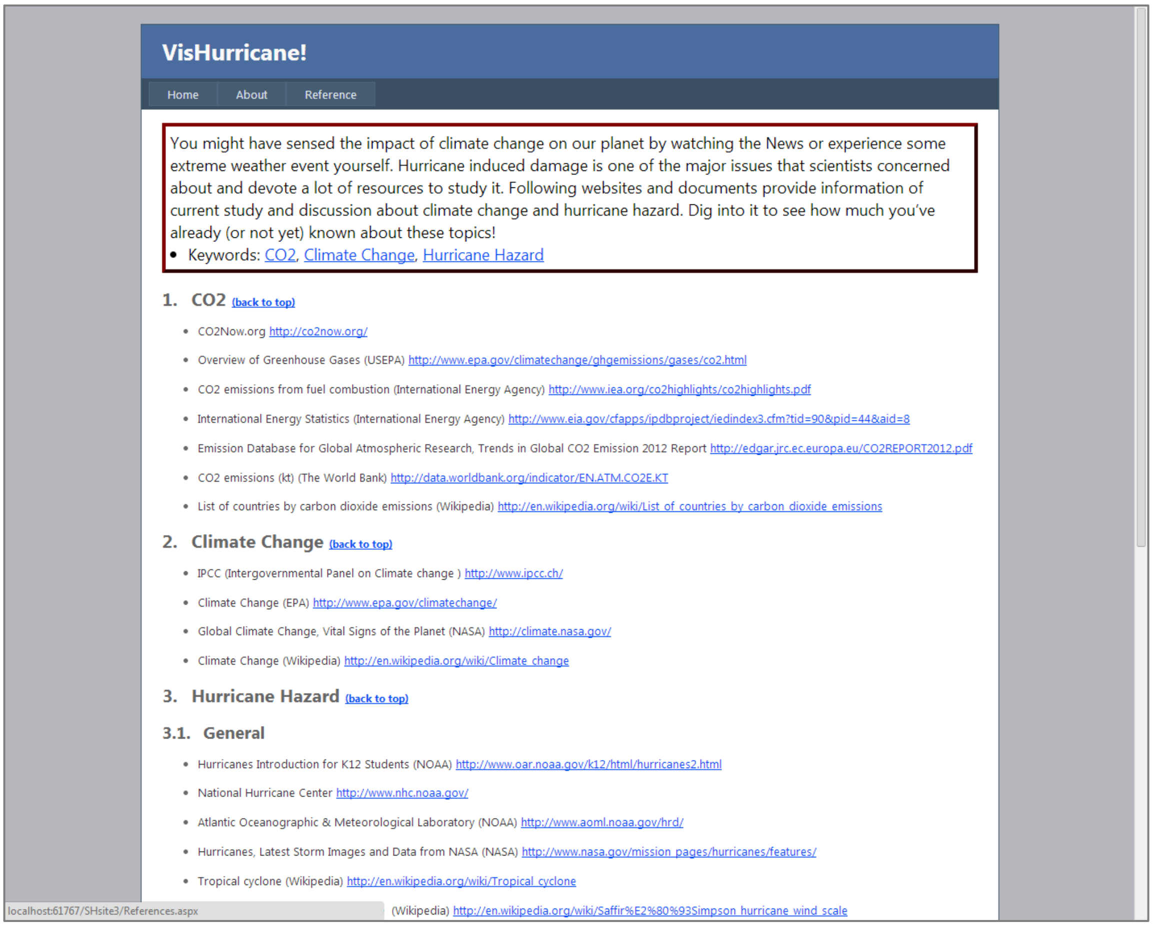
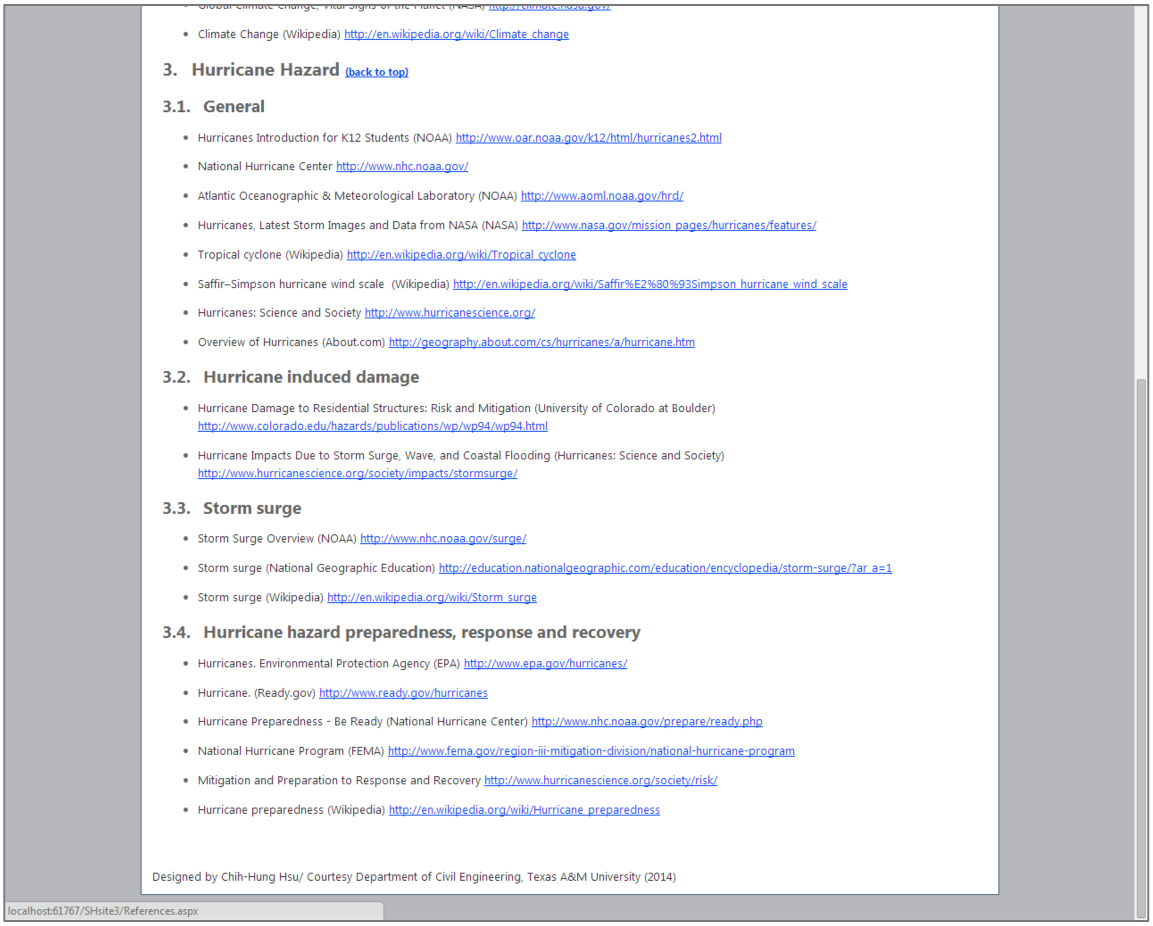


Figure 6-18 References.aspx(1)



- Climate Change (Wikipedia) http://en.wikipedia.org/wiki/Climate_change

3. Hurricane Hazard (back to top)

3.1. General

- Hurricanes Introduction for K12 Students (NOAA) http://www.oar.noaa.gov/k12/html/hurricanes2.html
- National Hurricane Center http://www.nhc.noaa.gov/
- Atlantic Oceanographic & Meteorological Laboratory (NOAA) http://www.aoml.noaa.gov/hrd/
- Hurricanes, Latest Storm Images and Data from NASA (NASA) http://www.nasa.gov/mission_pages/hurricanes/features/
- Tropical cyclone (Wikipedia) http://en.wikipedia.org/wiki/Tropical_cyclone
- Saffir–Simpson hurricane wind scale (Wikipedia) http://en.wikipedia.org/wiki/Saffir%E2%80%93Simpson_hurricane_wind_scale
- Hurricanes: Science and Society http://www.hurricanescience.org/
- Overview of Hurricanes (About.com) http://geography.about.com/cs/hurricanes/a/hurricane.htm

3.2. Hurricane induced damage

- Hurricane Damage to Residential Structures: Risk and Mitigation (University of Colorado at Boulder) http://www.colorado.edu/hazards/publications/wp/wp94/wp94.html
- Hurricane Impacts Due to Storm Surge, Wave, and Coastal Flooding (Hurricanes: Science and Society) http://www.hurricanescience.org/society/impacts/stormsurge/

3.3. Storm surge

- Storm Surge Overview (NOAA) http://www.nhc.noaa.gov/surge/
- Storm surge (National Geographic Education) http://education.nationalgeographic.com/education/encyclopedia/storm-surge/?ar_a=1
- Storm surge (Wikipedia) http://en.wikipedia.org/wiki/Storm_surge

3.4. Hurricane hazard preparedness, response and recovery

- Hurricanes. Environmental Protection Agency (EPA) http://www.epa.gov/hurricanes/
- Hurricane. (Ready.gov) http://www.ready.gov/hurricanes
- Hurricane Preparedness - Be Ready (National Hurricane Center) http://www.nhc.noaa.gov/prepare/ready.php
- National Hurricane Program (FEMA) http://www.fema.gov/region-iii-mitigation-division/national-hurricane-program
- Mitigation and Preparation to Response and Recovery http://www.hurricanescience.org/society/risk/
- Hurricane preparedness (Wikipedia) http://en.wikipedia.org/wiki/Hurricane_preparedness

Designed by Chih-Hung Hsu/ Courtesy Department of Civil Engineering, Texas A&M University (2014)

localhost:61767/SHsite3/References.aspx

Figure 6-19 References.aspx(2)

7. CONCLUSION

A framework has been developed to analysis hurricane flood risk at three study areas, namely Corpus Christ, TX, Gulfport, MS and Panama, FL. This approach is relatively efficient than traditional approach that needs to run hydrodynamic model for a great amount of hurricane scenarios. This study has investigated the hurricane flood risk and includes effects of sea level rise and hurricane intensification as a result of climate change. The results are displayed in terms of three future climate scenarios (B1, A1B and A1FI) at two time points (year 2030 and 2080) in the future.

The analysis of the SST effects on surges shows that under future climate conditions, the surge values will be increasing. The investigation on the surge value show that the surge value increases most significantly at Panama, FL. The highest expected surge increase could get as much as 1.5 m, while the value is 0.6 m for the other two cities. The expected surge increase at Corpus Christi shows that the increased surge value for locations inside the bay is less than locations around the barrier island for climate scenarios with higher SST, which indicates that the presence of the barrier islands play an important role on surge attenuation in the bay, as the barrier island avoid the surge that is generated in the open sea to propagate into the bay. This phenomenon is not observed for the other two cities where there is no such barrier island topography as there is in Corpus Christi.

The analysis of the SLR effects on damage shows that inundated property value in Gulfport is mostly sensitive to SLR among the three study areas as more property is distributed at low-lying area than the other cities. In 2030, the estimated inundated properties value ranges from 8.4 to 13 million at Corpus Christi; ranges from 17 to 18 million dollars Gulfport; ranges from 41 to 42 million dollars at Panama; in 2080, the inundated property value due to SLR ranges from 40 to 170 million dollars at Corpus Christi; ranges from 18 to 420 million at Gulfport; ranges from 42 to 52 million dollars at Panama. The affected population is mostly sensitive to SLR at Corpus Christi, as more people live in the low-lying area then the other cities. In 2030, the estimation of affected population ranges from 10 to 80 at Corpus Christi; ranges from 90 to 100 at Gulfport; ranges from 210 to 220 at Panama; in 2080, the estimation of affected population ranges from 200 to 1900 at Corpus Christi; from 100 to 770 at Gulfport; from 230 to 800 at Panama. The flooded area is mostly sensitive to SLR at Corpus Christi, as more area has low elevation than the other two cities. In 2030, the estimated flooded area ranges from 13 to 14 km^2 at Corpus Christi; from 4 to 5 km^2 at Gulfport; from 1.5 to 1.6 km^2 at Panama; in 2080, the estimated flooded area ranges from 16 to 38 km^2 at Corpus Christi; from 5 to 17 km^2 at Gulfport; from 3.8 to 21 km^2 at Panama.

The expected annual loss maps show that, the most vulnerable area of Corpus Christi is the north side and south side of the barrier island; the most vulnerable area of Gulfport, MS are the areas around Bay St. Louis and Biloxi Bay; and the most vulnerable area of Panama, FL are the areas close to St. Andrew Bay and the North Bay. As the map display the loss due to SLR as complete loss of the property value and displayed in red (100%

loss), the resulting maps show that SLR has the most significant effect at Corpus Christi, as more parcels are inundated at Corpus Christi than the other two cities. On the other hand, the increased surge value (due to increased SST and intensified hurricane) contribute more to the damage value at the other two cities, as more parcels show in yellow or orange than at Corpus Christi.

The damage-return period for parcel data show that the selected future climate scenarios at 2080 significantly increase the values relative to 2030. Take the 100 year event as example, under current climate condition, the damage are 50, 700 and 200 million dollars for Corpus Christi, Gulfport and Panama City, respectively; the affected population are 8,000, 30,000, and 15,000; the flooded area are 160, 130 and 180 km^2 for Corpus Christi, Gulfport and Panama City, also respectively. The future climate conditions could cause increases the damage value, affected population, and flooded area. At Corpus Christi, the damage (affected population, flooded area) could increase by 60~160% (30~60%, 13~25%) of the value under current climate condition for scenarios at 2030; and increase by 200~660% (75~190%, 30~60%) of the value under current climate condition for scenarios a 2080. At Gulfport, the damage (affected population, flooded area) could increase by 10~20% (13~26%, 12~24%) of the value under current climate condition for scenarios at 2030; and increase by 35~70% (30~100%, 45~80%) of the value under current climate condition for scenarios at 2080. At Panama City, the damage (affected population, flooded area) could increase by 15~50% (13~40%, 7~17%) of the value under current climate condition for scenarios at 2030; and increase by 50~250% (40~130%, 20~70%) of the value under current climate condition for scenarios at 2080.

The damage-return period for the entire business data show that, for 100-year event, under current climate condition, the affected number of business are 500, 2,100, and 700 for Corpus Christi, Gulfport and Panama City, respectively; the affected employee are 4,000, 30,000 and 4,000, the affected sales volume are 500, 4,000 and 700 million dollars for Corpus Christi, Gulfport and Panama City, also respectively. At Corpus Christi, the affected number of business (affected employee, affected sales volume) could increase by 80~160% (100~200%, 100~200%) of the value under current climate condition for scenarios at 2030; and increase by 180~280% (225~325%, 220~320%) of the value under current climate condition for scenarios a 2080. At Gulfport, the affected number of business (affected employee, affected sales volume) could increase by 14~48% (10~23%, 13~38%) of the value under current climate condition for scenarios at 2030; and increase by 48~114% (33~100%, 40~100%) of the value under current climate condition for scenarios at 2080. At Panama City, affected number of business (affected employee, affected sales volume) could increase by 14~43% (20~50%, 14~43%) of the value under current climate condition for scenarios at 2030; and increase by 30~160% (50~250%, 30~200%) of the value under current climate condition for scenarios at 2080.

The damage-return period analysis for the sub-categorized businesses (defined by 3-digit NAICS code) shows that different types of businesses are affected by hurricane surge flooding differently. Some variables of some businesses are sensitive to the effects of future climate conditions (e.g. the Sales volume and number of employee of “Specialty Trade Contractors ” at Corpus Christi in 2030 (Appendix G.4) ; the sales volume and number of employee of “Amusement, Gambling, and Recreation Industry” at Gulfport in 2030 (Appendix G.5); the sales volume, number of employee, and number of business of “Professional, Scientific, and Technical Services” at Panama in 2030 (Appendix G.12)), while some are not (e.g. most of the other variables of other businesses). The slope of the damage-return period curve and the increment due to changed climate condition all depend on the location and elevation of the business location. For example, the affected sales volume and number of employee for “Amusement, Gambling, and Recreation Industry” at Gulfport in 2080 is significantly affected by the sea level rise, because a major portion of the sales volume and number of employee belong to the data points below sea level in 2080 (see Figure 5-42).

The result of landfall location effect on damage shows that, under current climate condition, the hypothetical hurricane (c_p=950 mb, R_p=16 nm) would cause greater flood damage and affected the more population at Gulfport than at the other two cities. The same hurricane under future climate conditions in 2080 could significantly increase the damage at Corpus Christi (mostly by 10 times the value under current condition), while the effects on the other two cities are relatively smaller.

The investigation of the estimation error on flood damage, affected population and flooded area under current climate condition show that, for the selected hurricane scenario (cp=950 mb, Rp=16nm), the errors are approximately $\pm$ 100 million dollars for flood damage, approximately $\pm$ 5,000 for affected population and $\pm$ 10~50 km^2 for flooded area, assuming the surge values at the SRF stations are $\pm 40\ cm$ off the value obtained from surge response values.

The sensitively analysis for hurricane parameters on the damage caused by the hypothetical hurricane (cp=950 mb, Rp=16 nm) shows that, hurricane central pressure and radius to the maximum wind are the variables that affects the damage estimation most significantly. In general lower hurricane central pressure, larger radius to the maximum, negative approach angle and greater forward speed would have positive contribution to the damage values. While most parameters only change the magnitude of the damage value, the difference in radius to the maximum wind could change the landfall location where the hurricane causes the greatest damage.

The comparison of the damage estimation for Corpus Christi between this study and Frey et al. (2010) shows that, while the estimation form this study are lower in some cases (estimation of damage at current time and 2030; estimation of affected population) and greater in others (estimation of damage at 2080; estimation of flooded area), the differences are within order of three.

The flood maps for the hypothetical hurricane (with 950 mb central pressure and 16 nm radius) making landfall at locations which cause the greatest damage for the three cities (under current climate condition) show that, Gulfport is subject to greater flood depth (up to 2~2.5m) than the other two cities, which coincide with the result of the landfall location effect plots (Figure 5-46 through Figure 5-48) on damage estimation. The area where gets flooded at the three cities (under this “worst scenario”) also match the areas that are the most vulnerable on the expected annual loss maps.

In summary, the presented framework, which incorporated with surge response functions, can efficiently derive the value of risk and damage, with consideration of future climate conditions (SLR,ΔSST), with only limited runs of hydrodynamic model. The produced risk maps, flood maps and damage values for extreme events can provide information to support decision making on hazard management. Future implementation of the framework could include population dynamics in order to more accurately estimate flood damage and affected population due to hurricane flood in the future. To more thoroughly evaluate the damage induced by hurricane events, one should also include the damage caused by wind and inland flooding, in which the inland flood modeling and wind damage modeling should be incorporated. For example, overlying the flood areas from inland flooding and surge flooding would leads to more accurately approximation of affected population and property loss.

A web-based game, “VisHurricane” is developed based on the concept of the “parameterized hurricane flood damage analysis” in this study, in which the flood damage, affected population and flooded area along with hurricane parameters are tabulated and stored in a database, so that the damage value can be retrieved when given hurricane intensity and size. Hurricane events specified by the hurricane intensity and size are simulated in VisHurricane. The change of hurricane intensity is simulated based on Knutson and Tuleya (2004) which assumes that SST increase would cause hurricane intensification, hence result in different amount of damage. “VisHuricane” provides information (documents and websites) for K12 students to get to understand the issue of hurricane flood damage and the effect of climate change on the damage values, also get a sense of how the uncertainty play a role on hurricane surge flood events and induced damage. Further improvement of “VisHurricane” could include relocation of population and properties or other damage mitigation measures, to enrich the contents and allow the students to understand more comprehensively about hurricane hazard management process in real case.

REFERENCES

Baptista, A. M., Y. Zhang, A. Chawla, M. Zulauf, C. Seaton, E. P. Myers Iii, J. Kindle, M. Wilkin, M. Burla and P. J. Turner, 2005. A Cross-Scale Model for 3D Baroclinic Circulation in Estuary-Plume-Shelf Systems: II. Application to the Columbia River. *Continental Shelf Research* **25**:935-972.

Bjarnadottir, S., Y. Li and M. Stewart, 2011. Social Vulnerability Index for Coastal Communities at Risk To Hurricane Hazard And A Changing Climate. *Natural Hazards* **59**:1055-1075.

Bjarnadottir, S., Y. Li and M. G. Stewart, 2013. Regional Loss Estimation Due To Hurricane Wind and Hurricane-Induced Surge Considering Climate Variability. *Structure and Infrastructure Engineering* 10.1080/15732479.2013.816973:1-16.

Bunya, S., J. C. Dietrich, J. J. Westerink, B. A. Ebersole, J. M. Smith, J. H. Atkinson, R. Jensen, D. T. Resio, R. A. Luettich, C. Dawson, V. J. Cardone, A. T. Cox, M. D. Powell, H. J. Westerink and H. J. Roberts, 2010. A High-Resolution Coupled Riverine Flow, Tide, Wind, Wind Wave, and Storm Surge Model for Southern Louisiana and Mississippi. Part I: Model Development and Validation. *Monthly Weather Review* **138**:345-377.

Chen, C., H. Liu and R. C. Beardsley, 2003. An Unstructured Grid, Finite-Volume, Three-Dimensional, Primitive Equations Ocean Model: Application to Coastal Ocean and Estuaries. *Journal of Atmospheric & Oceanic Technology* **20**:159.

Chen, Q., L. Wang, H. Zhao and S. L. Douglass, 2007. Prediction of Storm Surges and Wind Waves on Coastal Highways in Hurricane-Prone Areas. *Journal of Coastal Research* **23**:1304-1317.

Church, J., N. White, T. Aarup, W. S. Wilson, P. Woodworth, C. Domingues, J. Hunter and K. Lambeck, 2008. Understanding Global Sea Levels: Past, Present and Future. *Sustain Sci* **3**:9-22.

Church, J. A., N. J. White, R. Coleman, K. Lambeck and J. X. Mitrovica, 2004. Estimates of the Regional Distribution of Sea Level Rise over the 1950–2000 Period. *Journal of Climate* **17**:2609-2625.

Cooper, M. P., M. Beevers and M. Oppenheimer, 2008. The Potential Impacts of Sea Level Rise on the Coastal Region of New Jersey, USA. *Climatic Change*

90:475-492.

Dietrich, J. C., S. Bunya, J. J. Westerink, B. A. Ebersole, J. M. Smith, J. H. Atkinson, R. Jensen, D. T. Resio, R. A. Luettich, C. Dawson, V. J. Cardone, A. T. Cox, M. D. Powell, H. J. Westerink and H. J. Roberts, 2010. A High-Resolution Coupled Riverine Flow, Tide, Wind, Wind Wave, and Storm Surge Model for Southern Louisiana and Mississippi. Part II: Synoptic Description and Analysis of Hurricanes Katrina and Rita. *Monthly Weather Review* **138**:378-404.

Dietrich, J. C., M. Zijlema, J. J. Westerink, L. H. Holthuijsen, C. Dawson, R. A. Luettich Jr, R. E. Jensen, J. M. Smith, G. S. Stelling and G. W. Stone, 2011. Modeling Hurricane Waves and Storm Surge Using Integrally-Coupled, Scalable Computations. *Coastal Engineering* **58**:45-65.

Ebersole, B. A., J. J. Westerink, S. Bunya, J. C. Dietrich and M. A. Cialone, 2010. Development of Storm Surge Which Led To Flooding in St. Bernard Polder during Hurricane Katrina. *Ocean Engineering* **37**:91-103.

Elsner, J. B., 2006. Evidence In Support Of the Climate Change-Atlantic Hurricane Hypothesis. *Geophysical Research Letters* **33**:L16705.

Elsner, J. B., J. P. Kossin and T. H. Jagger, 2008. The Increasing Intensity of the Strongest Tropical Cyclones. *Nature* **455**:92-95.

Emanuel, K., 2005. Increasing Destructiveness of Tropical Cyclones Over the Past 30 Years. *Nature* **436**:686-688.

FEMA (Federal Emergency Management Agency), 2014. Costal Analysis and Mapping. http://www.region2coastal.com/coastal-mapping-basics. accessed 2014.

Ferreira, C. M., 2012. The Influence of Coastal Wetlands on Hurricane Surge And Damage With Application To Planning Under Climate Change, Texas A&M University.

Frazier, T. G., N. Wood, B. Yarnal and D. H. Bauer, 2010. Influence of Potential Sea Level Rise on Societal Vulnerability to Hurricane Storm-Surge Hazards, Sarasota County, Florida. *Applied Geography* **30**:490-505.

Frey, A. E., F. Olivera, J. L. Irish, L. M. Dunkin, J. M. Kaihatu, C. M. Ferreira and B. L. Edge, 2010. Potential Impact of Climate Change on Hurricane Flooding Inundation, Population Affected and Property Damages in Corpus Christi.

Journal of the American Water Resources Association **46**:1049-1059.

Fritz, H., C. Blount, R. Sokoloski, J. Singleton, A. Fuggle, B. McAdoo, A. Moore, C. Grass and B. Tate, 2008. Hurricane Katrina Storm Surge Reconnaissance. *Journal of Geotechnical and Geoenvironmental Engineering* **134**:644-656.

Gee, J. P., 2004. *What Video Games Have to Teach Us About Learning and Literacy*. New York, Palgrave Macmillan,

GRID-Arendal, 2013. IPCC Third Assessment Report, Appendix II SRES Tables. http://www.grida.no/publications/other/ipcc_tar/. accessed Jan 2013.

Gulport MS, 2012. Gulfport, Mississippi. http://www.gulfport-ms.gov/download.shtml. accessed 2012.

Hinrichsen, D., 1999. The Coastal Population Explosion. *In*: *Trends and Future Challenges for U.S. National Ocean and Coastal Policy.* NOAA, Washington, D.C.

Ho, F. P. and V. A. Myers, 1975. Joint Probability Method of Tide Frequency Analysis Applied to Apalachicola Bay and St. George Sound, Florida. *In: NOAA Tech. Rep. NWS*. p. p43.

Huang, Y., R. H. Weisberg and L. Zheng, 2010. Coupling Of Surge and Waves for an Ivan-Like Hurricane Impacting the Tampa Bay, Florida region. *Journal of Geophysical Research: Oceans* **115**:C12009.

HURDAT, 2014. The North Atlantic Hurricane Database. http://www.nhc.noaa.gov/data/#hurdat. accessed 2014 March.

Infogroup, 2012. ReferenceUSA. http://www.referenceusa.com/Home/Home. accessed 2012.

IPCC, 2013. Data Distribuiton Centre, "Carbon Dioxide: Projected emissions and concentrations". http://www.ipcc-data.org/observ/ddc_co2.html. accessed Jan 2013.

IPCC, 2014a. Intergovernmental Panel on Climate Change. https://www.ipcc.ch/index.htm. accessed 2013, April.

IPCC, 2014b. Summary for Policymakers - A report of Working Group I of the Intergovernmental Panel on Climate Change. http://www.ipcc.ch/pdf/assessment-report/ar4/wg1/ar4-wg1-spm.pdf. accessed 2013, June.

IPCC (Intergovernmental Panel On Climate Change), 2007a. Climate Change 2007: The Physical Science Basis.

IPCC (Intergovernmental Panel On Climate Change), 2007b. Contribution of working group I to the fourth assessment report of the intergovernmental panel on climate change. *In*: *Climate Change 2007: The physical science basis.*, S. Solomon, D. Qin, M. Manning, Z. Chen, M. Marquis, K.B. Averyt, M. Tignor and H. L. Miller. Cambridge University Press, Cambridge, pp. 1-996.

Irish, J., A. Frey, M. E. Mousavi, F. Olivera and B. Edge, 2008a. Estimating the Influence of Projected Global Warming Scenarios on Hurricane Flooding.

Irish, J. and D. Resio, 2013. Method for Estimating Future Hurricane Flood Probabilities and Associated Uncertainty. *Journal of Waterway, Port, Coastal, and Ocean Engineering* **139**:126-134.

Irish, J., D. Resio and M. Cialone, 2009. A Surge Response Function Approach to Coastal Hazard Assessment - Part 2: Quantification of Spatial Attributes Of Response Functions. *Natural Hazards* **51**:183-205.

Irish, J., D. Resio and J. Ratcliff, 2008b. The Influence of Storm Size on Hurricane Surge. *Journal of Physical Oceanography* **38**:2003-2013.

Irish, J., Y. K. Song and K.-A. Chang, 2011a. Probabilistic Hurricane Surge Forecasting Using Parameterized Surge Response Functions. *Geophys. Res. Lett.* **38**:L03606.

Irish, J. L., C. Ferreira, F. Olivera, D. T. Resio and C.-H. Hsu, 2012. Hurricane Hazard Assessment: Considerations for Sea-Level Rise and Climate Change. *Proceedings of ... Conference on Coastal Engineering* **1**.

Irish, J. L., D. T. Resio and D. Divoky, 2011b. Statistical Properties of Hurricane Surge along a Coast. *Journal of Geophysical Research: Oceans* **116**:C10007.

Knutson, T. R. and R. E. Tuleya, 2004. Impact of CO2-Induced Warming on Simulated Hurricane Intensity and Precipitation: Sensitivity to the Choice of Climate Model and Convective Parameterization. *Journal of Climate* **17**:3477-3495.

Landsea, C. W., C. Anderson, N. Charles, G. Clark, J. Femandez-Partagas, P. Hungerford, C. Neumann and M. Zimmer, 2013. The Atlantic Hurricane Database Re-analysis Project, National Oceanic and Atmospheric Administration Report. http://www.aoml.noaa.gov/hrd/hurdat/Documentation.html. accessed 2013.

Li, Y., J. van de Lindt, T. Dao, S. Bjarnadottir and A. Ahuja, 2012. Loss Analysis for Combined Wind and Surge in Hurricanes. *Natural Hazards Review* **13**:1-10.

Lott, N. and T. Ross, 2006. Tracking and Evaluating U.S. Billion Dollar Weather Disasters 1980-2005. National Climatic Data Center (NCDC).

Luettich, R. and J. Westerink, 2004. Formulation and Numerical Implementation of a 2D/3D ADCIRC Finite Element Model Version. http://www.unc.edu/ims/adcirc/adcirc_theory_2004_12_08.pdf. accessed 2013.

Luo, J., A. P. Makwana, D. Liao and J. P. Kincaid, 2008. HURRICANE!-A Simulation-Based Program for Science Education. *In*: *2008 Winter Simulation Conference.*

Mattocks, C. and C. Forbes, 2008. A Real-Time, Event-Triggered Storm Surge Forecasting System for the State Of North Carolina. *Ocean Modelling* **25**:95-119.

Mousavi, M. E., J. L. Irish, A. E. Frey, F. Olivera and B. L. Edge, 2011. Global Warming and Hurricanes: The Potential Impact of Hurricane Intensification and Sea Level Rise on Coastal Flooding. *Climatic Change* **104**:575-597.

Myers, V. A., 1975. Storm Tide Frequencies on The South Carolina Coast. *In: NOAA Tech. Rep. NWS*. p. p79.

Nadal, N. C., R. E. Zapata, I. Pagán, R. López and J. Agudelo, 2010. Building Damage due to Riverine and Coastal Floods. *Journal of Water Resources Planning & Management* **136**:327-336.

Natioal Geographic, 2012. The Eye of the Hurricane. http://www.nationalgeographic.com/xpeditions/lessons/07/g35/trythishurricane.html. accessed 2012.

National Research Council, 2012. Sea-Level Rise for the Coasts of California, Oregon, and Washington. *In*., Washington, D.C.

Nicholls, R. and A. Cazenave, 2010. Sea-Level Rise and Its Impact on Coastal Zones.

Science **328**:1517-1520.

Niedoroda, A. W., D. T. Resio, G. Toro, D. Divoky and C. Reed, 2008. Efficient strategies for the joint probability evaluation of storm surge hazards. *In*: *Proceedings of the ASCE conference on Solutions to Coastal Disasters.* Turtle Bay, Hawaii, pp. pp 242–255.

Niedoroda, A. W., D. T. Resio, G. R. Toro, D. Divoky, H. S. Das and C. W. Reed, 2010. Analysis of the Coastal Mississippi Storm Surge Hazard. *Ocean Engineering* **37**:82-90.

NOAA, 2013. Tropical Cyclone Names (Atlantic). http://www.nhc.noaa.gov/aboutnames.shtml. accessed Jan 2013.

NOAA, 2014. Sea Level Is Rising at an Increasing Rate. http://oceanservice.noaa.gov/facts/sealevel.html. accessed 2014.

NOAA (National Oceanic and Atmospheric Administration), 2012. National Oceanic and Atmospheric Administration, Center for Operational Oceanographic Products and Services. http://tidesandcurrents.noaa.gov/. accessed 2012.

NOAA (National Oceanic and Atmospheric Administration), 2014. Sea, Lake, and Overland Surges from Hurricanes (SLOSH). http://www.nhc.noaa.gov/surge/slosh.php. accessed 2012.

Nueces County TX, 2012. Nueces County Apprisal District. http://www.ncadistrict.com/downloads.htm. accessed 2012.

Panama FL, 2012. Panama, Florida. http://www.co.bay.fl.us/gis/county.php. accessed 2012.

Rego, J. L. and C. Li, 2010. Storm Surge Propagation in Galveston Bay during Hurricane Ike. *Journal of Marine Systems* **82**:265-279.

Resio, D., J. Irish and M. Cialone, 2009. A Surge Response Function Approach to Coastal Hazard Assessment - Part 1: Basic Concepts. *Natural Hazards* **51**:163-182.

Resio, D., J. Irish, J. Westerink and N. Powell, 2013. The Effect of Uncertainty on Estimates of Hurricane Surge Hazards. *Natural Hazards* **66**:1443-1459.

Resio, D. and J. Westerink, 2008. Modeling the Physics of Storm Surges. *Physics Today* **61**:33-38.

Robertson, I., H. Riggs, S. Yim and Y. Young, 2007. Lessons from Hurricane Katrina Storm Surge on Bridges and Buildings. *Journal of Waterway, Port, Coastal, and Ocean Engineering* **133**:463-483.

San Patricio County TX, 2012. San Patricio County Appraisal District. http://www.spcad-egov.org/. accessed 2012.

Scawthorn, C., N. Blais, H. Seligson, E. Tate, E. Mifflin, W. Thomas, J. Murphy and C. Jones, 2006. HAZUS-MH Flood Loss Estimation Methodology. I: Overview and Flood Hazard Characterization. *Natural Hazards Review* **7**:60-71.

Schneider, P. and B. Schauer, 2006. HAZUS—Its Development and Its Future. *Natural Hazards Review* **7**:40-44.

Sheng, Y. P., Y. Zhang and V. A. Paramygin, 2010. Simulation of Storm Surge, Wave, and Coastal Inundation in the Northeastern Gulf Of Mexico Region during Hurricane Ivan In 2004. *Ocean Modelling* **35**:314-331.

Shepard, C., V. Agostini, B. Gilmer, T. Allen, J. Stone, W. Brooks and M. Beck, 2012. Assessing Future Risk: Quantifying the Effects of Sea Level Rise on Storm Surge Risk for the Southern Shores of Long Island, New York. *Natural Hazards* **60**:727-745.

Smith, J. M., M. A. Cialone, T. V. Wamsley and T. O. McAlpin, 2010. Potential Impact of Sea Level Rise on Coastal Surges in Southeast Louisiana. *Ocean Engineering* **37**:37-47.

Song, Y., J. Irish and I. Udoh, 2012. Regional Attributes of Hurricane Surge Response Functions for Hazard Assessment *Natural Hazards* 10.1007/s11069-012-0309-z:1-16.

Toro, G. R., D. T. Resio, D. Divoky, A. W. Niedoroda and C. Reed, 2010. Efficient Joint-Probability Methods for Hurricane Surge Frequency Analysis. *Ocean Engineering* **37**:125-134.

U.S. Census Bureau, 2013a. 2010 Geographic Terms and Concepts. http://www.census.gov/geo/reference/terms.html. accessed 2013.

U.S. Census Bureau, 2013b. North American Industry Classification System. http://www.census.gov/eos/www/naics/. accessed 2013.

U.S. Census Bureau, 2013c. State & County QuickFacts. http://quickfacts.census.gov/qfd/. accessed 2013.

U.S. Census Bureau, 2013d. United States Census TIGER database. http://www.census.gov/geo/maps-data/data/tiger.html. accessed 2013.

UCAR, 2012. Univeristy Corporation for Atmosphere, Hurricane Strike!-An Interactive Course in Hurricane Science and Safety. https://www.meted.ucar.edu/training_module.php?id=31. accessed 2012.

Udoh, I. E., 2012. Robust hurricane Surge Response Functions, Dissertation, Texas A&M University.

USACE (U.S. Army Corps of Engineers), 2014. CH3D Numerical Modeling System. http://chl.erdc.usace.army.mil/chl.aspx?p=s&a=Software;22. accessed 2013.

USGS 2005. U.S. Geological Survey, Hurricane Hazards-A National Threat. Fact Sheet 2005-3121.

USGS, 2012. U.S. Geological Survey, National Elevation Dataset. http://ned.usgs.gov/. accessed 2012.

USGS, 2013. U.S. Geological Survey, The National Map Viewer and Download Platform. http://nationalmap.gov/viewer.html. accessed 2010.

van de Lindt, J., A. Graettinger, R. Gupta, T. Skaggs, S. Pryor and K. Fridley, 2007. Performance of Wood-Frame Structures during Hurricane Katrina. *Journal of Performance of Constructed Facilities* **21**:108-116.

Westerink, J. J., R. A. Luettich, J. C. Feyen, J. H. Atkinson, C. Dawson, H. J. Roberts, M. D. Powell, J. P. Dunion, E. J. Kubatko and H. Pourtaheri, 2008. A Basin- to Channel-Scale Unstructured Grid Hurricane Storm Surge Model Applied to Southern Louisiana. *Monthly Weather Review* **136**:833-864.

Wigley, T. M. L., 2008. MAGICC/SCENGEN 5.3: USER MANUAL (Version 2). http://www.cgd.ucar.edu/cas/wigley/magicc/UserMan5.3.v2.pdf. accessed 2014.

Wilson, S. G. and T. R. Fischetti, 2010. Coastline Population Trends in the United States: 1960 to 2008 - Population Estimates and Projections (US Census Bureau Report). http://www.census.gov/prod/2010pubs/p25-1139.pdf. accessed March 2013.

Wu, S.-Y., B. Yarnal and A. Fisher, 2002. Vulnerability of Coastal Communities to Sea-Level Rise: A Case Study of Cape May County, New Jersey, USA. *Climate Research* **22**:255-270.

Xu, H., K. Zhang, J. Shen and Y. Li, 2010. Storm Surge Simulation along the U.S. East and Gulf Coasts Using A Multi-Scale Numerical Model Approach. *Ocean Dynamics* **60**:1597-1619.

Yazdani, N., R. Dowgul and T. Manzur, 2010. Deficiency Analysis of Coastal Buildings toward Storm Damage Reduction. *Journal of Performance of Constructed Facilities* **24**:128-137.

Zhang, Y., A. M. Baptista and E. P. Myers, 2004. A Cross-Scale Model for 3D Baroclinic Circulation in Estuary-Plume-Shelf Systems: I. Formulation and Skill Assessment. *Continental Shelf Research* **24**:2187-2214.

APPENDIX A HAZUS DAMAGE CURVES TYPE (A ZONE)

Occupancy	**SpecificOccupId**	**Source**	**Description**	**Stories**	**ID**
RES1	R11N	FIA	one floor, no basement, Structure, A-Zone	1 Story	105
RES1	R11B	FIA (MOD.)	one floor, w/ basement, Structure, A-Zone	1 Story	106
RES1	R12N	FIA	two floors, no basement, Structure, A-Zone	2 Story	107
RES1	R12B	FIA (MOD.)	two floors, w/ basement, Structure, A-Zone	2 Story	108
RES1	R13N	FIA	three or more floors, no basement, Structure, A-Zone	3 Story	109
RES1	R13B	FIA (MOD.)	three or more floors, w/ basement, Structure, A-Zone	3 Story	110
RES1	R1SN	FIA	split level, no basement, Structure, A-Zone	Split Level	111
RES1	R1SB	FIA (MOD.)	split level, w/ basement, Structure, A-Zone	Split Level	112
RES2	R21N	FIA	Mobile home, structure, A-Zone	1 Story	189
RES2	R21B	FIA	Mobile home, structure, A-Zone	1 Story	189
RES3A	R3A1N	USACE - Chicago	Apartment Unit Grade, Structure	1to2 Stories	204
RES3A	R3A1B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	1to2 Stories	205
RES3A	R3A3N	USACE - Chicago	Apartment Unit Grade, Structure	3to4 Stories	204
RES3A	R3A3B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	3to4 Stories	205
RES3A	R3A5N	USACE - Chicago	Apartment Unit Grade, Structure	5Plus Stories	204
RES3A	R3A5B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	5Plus Stories	205
RES3B	R3B1N	USACE - Chicago	Apartment Unit Grade, Structure	1to2 Stories	204
RES3B	R3B1B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	1to2 Stories	205
RES3B	R3B3N	USACE - Chicago	Apartment Unit Grade, Structure	3to4 Stories	204
RES3B	R3B3B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	3to4 Stories	205
RES3B	R3B5N	USACE - Chicago	Apartment Unit Grade, Structure	5Plus Stories	204
RES3B	R3B5B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	5Plus Stories	205
RES3C	R3C1N	USACE - Chicago	Apartment Unit Grade, Structure	1to2 Stories	204
RES3C	R3C1B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	1to2 Stories	205
RES3C	R3C3N	USACE - Chicago	Apartment Unit Grade, Structure	3to4 Stories	204
RES3C	R3C3B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	3to4 Stories	205
RES3C	R3C5N	USACE - Chicago	Apartment Unit Grade, Structure	5Plus Stories	204

Occupancy	SpecificOccupId	Source	Description	Stories	ID
RES3C	R3C5B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	5Plus Stories	205
RES3D	R3D1N	USACE - Chicago	Apartment Unit Grade, Structure	1to2 Stories	204
RES3D	R3D1B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	1to2 Stories	205
RES3D	R3D3N	USACE - Chicago	Apartment Unit Grade, Structure	3to4 Stories	204
RES3D	R3D3B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	3to4 Stories	205
RES3D	R3D5N	USACE - Chicago	Apartment Unit Grade, Structure	5Plus Stories	204
RES3D	R3D5B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	5Plus Stories	205
RES3E	R3E1N	USACE - Chicago	Apartment Unit Grade, Structure	1to2 Stories	204
RES3E	R3E1B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	1to2 Stories	205
RES3E	R3E3N	USACE - Chicago	Apartment Unit Grade, Structure	3to4 Stories	204
RES3E	R3E3B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	3to4 Stories	205
RES3E	R3E5N	USACE - Chicago	Apartment Unit Grade, Structure	5Plus Stories	204
RES3E	R3E5B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	5Plus Stories	205
RES3F	R3F1N	USACE - Chicago	Apartment Unit Grade, Structure	1to2 Stories	204
RES3F	R3F1B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	1to2 Stories	205
RES3F	R3F3N	USACE - Chicago	Apartment Unit Grade, Structure	3to4 Stories	204
RES3F	R3F3B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	3to4 Stories	205
RES3F	R3F5N	USACE - Chicago	Apartment Unit Grade, Structure	5Plus Stories	204
RES3F	R3F5B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	5Plus Stories	205
RES4	R4LN	USACE - Galveston	Average Hotel & Motel, structure	Low Rise	209
RES4	R4LB	USACE - Galveston	Average Hotel & Motel, structure	Low Rise	209
RES4	R4MN	USACE - Galveston	Average Hotel & Motel, structure	Mid Rise	209
RES4	R4MB	USACE - Galveston	Average Hotel & Motel, structure	Mid Rise	209
RES4	R4HN	USACE - Galveston	Average Hotel & Motel, structure	High Rise	209
RES4	R4HB	USACE - Galveston	Average Hotel & Motel, structure	High Rise	209

Occupancy	SpecificOccupId	Source	Description	Stories	ID
RES5	R5LN	USACE - Galveston	Nursing Home, structure	Low Rise	214
RES5	R5LB	USACE - Galveston	Nursing Home, structure	Low Rise	214
RES5	R5MN	USACE - Galveston	Nursing Home, structure	Mid Rise	214
RES5	R5MB	USACE - Galveston	Nursing Home, structure	Mid Rise	214
RES5	R5HN	USACE - Galveston	Nursing Home, structure	High Rise	214
RES5	R5HB	USACE - Galveston	Nursing Home, structure	High Rise	214
RES6	R6LN	USACE - Galveston	Nursing Home, structure	Low Rise	215
RES6	R6LB	USACE - Galveston	Nursing Home, structure	Low Rise	215
RES6	R6MN	USACE - Galveston	Nursing Home, structure	Mid Rise	215
RES6	R6MB	USACE - Galveston	Nursing Home, structure	Mid Rise	215
RES6	R6HN	USACE - Galveston	Nursing Home, structure	High Rise	215
RES6	R6HB	USACE - Galveston	Nursing Home, structure	High Rise	215
COM1	C1LN	USACE - Galveston	Average Retail, Structure	Low Rise	217
COM1	C1LB	USACE - Galveston	Average Retail, Structure	Low Rise	217
COM1	C1MN	USACE - Galveston	Average Retail, Structure	Mid Rise	217
COM1	C1MB	USACE -	Average Retail, Structure	Mid Rise	217

Occupancy	SpecificOccupId	Source	Description	Stories	ID
		Galveston			
COM1	C1HN	USACE - Galveston	Average Retail, Structure	High Rise	217
COM1	C1HB	USACE - Galveston	Average Retail, Structure	High Rise	217
COM2	C2LN	USACE - Galveston	Average wholesale, Structure	Low Rise	341
COM2	C2LB	USACE - Galveston	Average wholesale, Structure	Low Rise	341
COM2	C2MN	USACE - Galveston	Average wholesale, Structure	Mid Rise	341
COM2	C2MB	USACE - Galveston	Average wholesale, Structure	Mid Rise	341
COM2	C2HN	USACE - Galveston	Average wholesale, Structure	High Rise	341
COM2	C2HB	USACE - Galveston	Average wholesale, Structure	High Rise	341
COM3	C3LN	USACE - Galveston	Average Personal & Repair Services, Structure	Low Rise	375
COM3	C3LB	USACE - Galveston	Average Personal & Repair Services, Structure	Low Rise	375
COM3	C3MN	USACE - Galveston	Average Personal & Repair Services, Structure	Mid Rise	375
COM3	C3MB	USACE - Galveston	Average Personal & Repair Services, Structure	Mid Rise	375
COM3	C3HN	USACE - Galveston	Average Personal & Repair Services, Structure	High Rise	375
COM3	C3HB	USACE - Galveston	Average Personal & Repair Services, Structure	High Rise	375
COM4	C4LN	USACE - Galveston	Average Prof/Tech Services, Structure	Low Rise	431

Occupancy	SpecificOccupId	Source	Description	Stories	ID
COM4	C4LB	USACE - Galveston	Average Prof/Tech Services, Structure	Low Rise	431
COM4	C4MN	USACE - Galveston	Average Prof/Tech Services, Structure	Mid Rise	431
COM4	C4MB	USACE - Galveston	Average Prof/Tech Services, Structure	Mid Rise	431
COM4	C4HN	USACE - Galveston	Average Prof/Tech Services, Structure	High Rise	431
COM4	C4HB	USACE - Galveston	Average Prof/Tech Services, Structure	High Rise	431
COM5	C5LN	USACE - Galveston	Bank, Structure	Low Rise	467
COM5	C5LB	USACE - Galveston	Bank, Structure	Low Rise	467
COM5	C5MN	USACE - Galveston	Bank, Structure	Mid Rise	467
COM5	C5MB	USACE - Galveston	Bank, Structure	Mid Rise	467
COM5	C5HN	USACE - Galveston	Bank, Structure	High Rise	467
COM5	C5HB	USACE - Galveston	Bank, Structure	High Rise	467
COM6	C6LN	USACE - Galveston	Hospital, Structure	Low Rise	474
COM6	C6LB	USACE - Galveston	Hospital, Structure	Low Rise	474
COM6	C6MN	USACE - Galveston	Hospital, Structure	Mid Rise	474
COM6	C6MB	USACE - Galveston	Hospital, Structure	Mid Rise	474
COM6	C6HN	USACE -	Hospital, Structure	High Rise	474

Occupancy	SpecificOccupId	Source	Description	Stories	ID
		Galveston			
COM6	C6HB	USACE - Galveston	Hospital, Structure	High Rise	474
COM7	C7LN	USACE - Galveston	Average Medical Office, Structure	Low Rise	475
COM7	C7LB	USACE - Galveston	Average Medical Office, Structure	Low Rise	475
COM7	C7MN	USACE - Galveston	Average Medical Office, Structure	Mid Rise	475
COM7	C7MB	USACE - Galveston	Average Medical Office, Structure	Mid Rise	475
COM7	C7HN	USACE - Galveston	Average Medical Office, Structure	High Rise	475
COM7	C7HB	USACE - Galveston	Average Medical Office, Structure	High Rise	475
COM8	C8LN	USACE - Galveston	Average Entertainment/Recreation, Structure	Low Rise	493
COM8	C8LB	USACE - Galveston	Average Entertainment/Recreation, Structure	Low Rise	493
COM8	C8MN	USACE - Galveston	Average Entertainment/Recreation, Structure	Mid Rise	493
COM8	C8MB	USACE - Galveston	Average Entertainment/Recreation, Structure	Mid Rise	493
COM8	C8HN	USACE - Galveston	Average Entertainment/Recreation, Structure	High Rise	493
COM8	C8HB	USACE - Galveston	Average Entertainment/Recreation, Structure	High Rise	493
COM9	C9LN	USACE - Galveston	Average theatre, Structure	Low Rise	532
COM9	C9LB	USACE - Galveston	Average theatre, Structure	Low Rise	532

Occupancy	SpecificOccupId	Source	Description	Stories	ID
COM9	C9MN	USACE - Galveston	Average theatre, Structure	Mid Rise	532
COM9	C9MB	USACE - Galveston	Average theatre, Structure	Mid Rise	532
COM9	C9HN	USACE - Galveston	Average theatre, Structure	High Rise	532
COM9	C9HB	USACE - Galveston	Average theatre, Structure	High Rise	532
COM10	C10LN	USACE - Galveston	Garage, structure	Low Rise	543
COM10	C10LB	USACE - Galveston	Garage, structure	Low Rise	543
COM10	C10MN	USACE - Galveston	Garage, structure	Mid Rise	543
COM10	C10MB	USACE - Galveston	Garage, structure	Mid Rise	543
COM10	C10HN	USACE - Galveston	Garage, structure	High Rise	543
COM10	C10HB	USACE - Galveston	Garage, structure	High Rise	543
IND1	I1LN	USACE - Galveston	Average heavy industrial, Structure	Low Rise	545
IND1	I1LB	USACE - Galveston	Average heavy industrial, Structure	Low Rise	545
IND1	I1MN	USACE - Galveston	Average heavy industrial, Structure	Mid Rise	545
IND1	I1MB	USACE - Galveston	Average heavy industrial, Structure	Mid Rise	545
IND1	I1HN	USACE - Galveston	Average heavy industrial, Structure	High Rise	545
IND1	I1HB	USACE -	Average heavy industrial, Structure	High Rise	545

Occupancy	SpecificOccupId	Source	Description	Stories	ID
		Galveston			
IND2	I2LN	USACE - Galveston	Average light industrial, structure	Low Rise	559
IND2	I2LB	USACE - Galveston	Average light industrial, structure	Low Rise	559
IND2	I2MN	USACE - Galveston	Average light industrial, structure	Mid Rise	559
IND2	I2MB	USACE - Galveston	Average light industrial, structure	Mid Rise	559
IND2	I2HN	USACE - Galveston	Average light industrial, structure	High Rise	559
IND2	I2HB	USACE - Galveston	Average light industrial, structure	High Rise	559
IND3	I3LN	USACE - Galveston	Average Food/Drug/Chem, Structure	Low Rise	575
IND3	I3LB	USACE - Galveston	Average Food/Drug/Chem, Structure	Low Rise	575
IND3	I3MN	USACE - Galveston	Average Food/Drug/Chem, Structure	Mid Rise	575
IND3	I3MB	USACE - Galveston	Average Food/Drug/Chem, Structure	Mid Rise	575
IND3	I3HN	USACE - Galveston	Average Food/Drug/Chem, Structure	High Rise	575
IND3	I3HB	USACE - Galveston	Average Food/Drug/Chem, Structure	High Rise	575
IND4	I4LN	USACE - Galveston	Average Metals/Minerals processing, structure	Low Rise	586
IND4	I4LB	USACE - Galveston	Average Metals/Minerals processing, structure	Low Rise	586
IND4	I4MN	USACE - Galveston	Average Metals/Minerals processing, structure	Mid Rise	586

Occupancy	SpecificOccupId	Source	Description	Stories	ID
IND4	I4MB	USACE - Galveston	Average Metals/Minerals processing, structure	Mid Rise	586
IND4	I4HN	USACE - Galveston	Average Metals/Minerals processing, structure	High Rise	586
IND4	I4HB	USACE - Galveston	Average Metals/Minerals processing, structure	High Rise	586
IND5	I5LN	USACE - Galveston	Average High Technology, structure	Low Rise	591
IND5	I5LB	USACE - Galveston	Average High Technology, structure	Low Rise	591
IND5	I5MN	USACE - Galveston	Average High Technology, structure	Mid Rise	591
IND5	I5MB	USACE - Galveston	Average High Technology, structure	Mid Rise	591
IND5	I5HN	USACE - Galveston	Average High Technology, structure	High Rise	591
IND5	I5HB	USACE - Galveston	Average High Technology, structure	High Rise	591
IND6	I6LN	USACE - Galveston	Average Construction, structure	Low Rise	592
IND6	I6LB	USACE - Galveston	Average Construction, structure	Low Rise	592
IND6	I6MN	USACE - Galveston	Average Construction, structure	Mid Rise	592
IND6	I6MB	USACE - Galveston	Average Construction, structure	Mid Rise	592
IND6	I6HN	USACE - Galveston	Average Construction, structure	High Rise	592
IND6	I6HB	USACE - Galveston	Average Construction, structure	High Rise	592
AGR1	A1LN	USACE -	Average Agriculture, structure	Low Rise	616

Occupancy	SpecificOccupId	Source	Description	Stories	ID
		Galveston			
AGR1	A1LB	USACE - Galveston	Average Agriculture, structure	Low Rise	616
AGR1	A1MN	USACE - Galveston	Average Agriculture, structure	Mid Rise	616
AGR1	A1MB	USACE - Galveston	Average Agriculture, structure	Mid Rise	616
AGR1	A1HN	USACE - Galveston	Average Agriculture, structure	High Rise	616
AGR1	A1HB	USACE - Galveston	Average Agriculture, structure	High Rise	616
REL1	RE1LN	USACE - Galveston	Church, structure	Low Rise	624
REL1	RE1LB	USACE - Galveston	Church, structure	Low Rise	624
REL1	RE1MN	USACE - Galveston	Church, structure	Mid Rise	624
REL1	RE1MB	USACE - Galveston	Church, structure	Mid Rise	624
REL1	RE1HN	USACE - Galveston	Church, structure	High Rise	624
REL1	RE1HB	USACE - Galveston	Church, structure	High Rise	624
GOV1	G1LN	USACE - Galveston	Average government services, structure	Low Rise	631
GOV1	G1LB	USACE - Galveston	Average government services, structure	Low Rise	631
GOV1	G1MN	USACE - Galveston	Average government services, structure	Mid Rise	631
GOV1	G1MB	USACE - Galveston	Average government services, structure	Mid Rise	631

Occupancy	SpecificOccupId	Source	Description	Stories	ID
GOV1	G1HN	USACE - Galveston	Average government services, structure	High Rise	631
GOV1	G1HB	USACE - Galveston	Average government services, structure	High Rise	631
GOV2	G2LN	USACE - Galveston	Average emergency response, Structure	Low Rise	640
GOV2	G2LB	USACE - Galveston	Average emergency response, Structure	Low Rise	640
GOV2	G2MN	USACE - Galveston	Average emergency response, Structure	Mid Rise	640
GOV2	G2MB	USACE - Galveston	Average emergency response, Structure	Mid Rise	640
GOV2	G2HN	USACE - Galveston	Average emergency response, Structure	High Rise	640
GOV2	G2HB	USACE - Galveston	Average emergency response, Structure	High Rise	640
EDU1	E1LN	USACE - Galveston	Average school, structure	Low Rise	643
EDU1	E1LB	USACE - Galveston	Average school, structure	Low Rise	643
EDU1	E1MN	USACE - Galveston	Average school, structure	Mid Rise	643
EDU1	E1MB	USACE - Galveston	Average school, structure	Mid Rise	643
EDU1	E1HN	USACE - Galveston	Average school, structure	High Rise	643
EDU1	E1HB	USACE - Galveston	Average school, structure	High Rise	643
EDU2	E2LN	USACE - Galveston	Average college/university, structure	Low Rise	652
EDU2	E2LB	USACE -	Average college/university, structure	Low Rise	652

Occupancy	SpecificOccupId	Source	Description	Stories	ID
		Galveston			
EDU2	E2MN	USACE - Galveston	Average college/university, structure	Mid Rise	652
EDU2	E2MB	USACE - Galveston	Average college/university, structure	Mid Rise	652
EDU2	E2HN	USACE - Galveston	Average college/university, structure	High Rise	652
EDU2	E2HB	USACE - Galveston	Average college/university, structure	High Rise	652

APPENDIX B HAZUS DAMAGE CURVE TYPES (V ZONE)

Occupancy	SpecificOccupId	Source	Description	Stories	ID
RES1	R11N	FIA	one floor, no basement, Structure, V-Zone	1 Story	113
RES1	R11B	FIA (MOD.)	one floor, w/ basement, Structure, V-Zone	1 Story	114
RES1	R12N	FIA	two floors, no basement, Structure, V-Zone	2 Story	115
RES1	R12B	FIA (MOD.)	two floors, w/ basement, Structure, V-Zone	2 Story	116
RES1	R13N	FIA	three or more floors, no basement, Structure, V-Zone	3 Story	117
RES1	R13B	FIA (MOD.)	three or more floors, w/ basement, Structure, V-Zone	3 Story	118
RES1	R1SN	FIA	split level, no basement, Structure, V-Zone	Split Level	119
RES1	R1SB	FIA (MOD.)	split level, w/ basement, Structure, V-Zone	Split Level	120
RES2	R21N	FIA	Mobile home, structure, V-Zone	1 Story	190
RES2	R21B	FIA	Mobile home, structure, V-Zone	1 Story	190
RES3A	R3A1N	USACE - Chicago	Apartment Unit Grade, Structure	1to2 Stories	204
RES3A	R3A1B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	1to2 Stories	205
RES3A	R3A3N	USACE - Chicago	Apartment Unit Grade, Structure	3to4 Stories	204
RES3A	R3A3B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	3to4 Stories	205
RES3A	R3A5N	USACE - Chicago	Apartment Unit Grade, Structure	5Plus Stories	204
RES3A	R3A5B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	5Plus Stories	205
RES3B	R3B1N	USACE - Chicago	Apartment Unit Grade, Structure	1to2 Stories	204
RES3B	R3B1B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	1to2 Stories	205
RES3B	R3B3N	USACE - Chicago	Apartment Unit Grade, Structure	3to4 Stories	204

Occupancy	SpecificOccupId	Source	Description	Stories	ID
RES3B	R3B3B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	3to4 Stories	205
RES3B	R3B5N	USACE - Chicago	Apartment Unit Grade, Structure	5Plus Stories	204
RES3B	R3B5B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	5Plus Stories	205
RES3C	R3C1N	USACE - Chicago	Apartment Unit Grade, Structure	1to2 Stories	204
RES3C	R3C1B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	1to2 Stories	205
RES3C	R3C3N	USACE - Chicago	Apartment Unit Grade, Structure	3to4 Stories	204
RES3C	R3C3B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	3to4 Stories	205
RES3C	R3C5N	USACE - Chicago	Apartment Unit Grade, Structure	5Plus Stories	204
RES3C	R3C5B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	5Plus Stories	205
RES3D	R3D1N	USACE - Chicago	Apartment Unit Grade, Structure	1to2 Stories	204
RES3D	R3D1B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	1to2 Stories	205
RES3D	R3D3N	USACE - Chicago	Apartment Unit Grade, Structure	3to4 Stories	204
RES3D	R3D3B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	3to4 Stories	205
RES3D	R3D5N	USACE - Chicago	Apartment Unit Grade, Structure	5Plus Stories	204
RES3D	R3D5B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	5Plus Stories	205
RES3E	R3E1N	USACE - Chicago	Apartment Unit Grade, Structure	1to2 Stories	204

Occupancy	SpecificOccupId	Source	Description	Stories	ID
RES3E	R3E1B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	1to2 Stories	205
RES3E	R3E3N	USACE - Chicago	Apartment Unit Grade, Structure	3to4 Stories	204
RES3E	R3E3B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	3to4 Stories	205
RES3E	R3E5N	USACE - Chicago	Apartment Unit Grade, Structure	5Plus Stories	204
RES3E	R3E5B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	5Plus Stories	205
RES3F	R3F1N	USACE - Chicago	Apartment Unit Grade, Structure	1to2 Stories	204
RES3F	R3F1B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	1to2 Stories	205
RES3F	R3F3N	USACE - Chicago	Apartment Unit Grade, Structure	3to4 Stories	204
RES3F	R3F3B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	3to4 Stories	205
RES3F	R3F5N	USACE - Chicago	Apartment Unit Grade, Structure	5Plus Stories	204
RES3F	R3F5B	USACE - Chicago	Apartment Unit Sub-Grade, Structure	5Plus Stories	205
RES4	R4LN	USACE - Galveston	Average Hotel & Motel, structure	Low Rise	209
RES4	R4LB	USACE - Galveston	Average Hotel & Motel, structure	Low Rise	209
RES4	R4MN	USACE - Galveston	Average Hotel & Motel, structure	Mid Rise	209
RES4	R4MB	USACE - Galveston	Average Hotel & Motel, structure	Mid Rise	209
RES4	R4HN	USACE - Galveston	Average Hotel & Motel, structure	High Rise	209

Occupancy	SpecificOccupId	Source	Description	Stories	ID
RES4	R4HB	USACE - Galveston	Average Hotel & Motel, structure	High Rise	209
RES5	R5LN	USACE - Galveston	Nursing Home, structure	Low Rise	214
RES5	R5LB	USACE - Galveston	Nursing Home, structure	Low Rise	214
RES5	R5MN	USACE - Galveston	Nursing Home, structure	Mid Rise	214
RES5	R5MB	USACE - Galveston	Nursing Home, structure	Mid Rise	214
RES5	R5HN	USACE - Galveston	Nursing Home, structure	High Rise	214
RES5	R5HB	USACE - Galveston	Nursing Home, structure	High Rise	214
RES6	R6LN	USACE - Galveston	Nursing Home, structure	Low Rise	215
RES6	R6LB	USACE - Galveston	Nursing Home, structure	Low Rise	215
RES6	R6MN	USACE - Galveston	Nursing Home, structure	Mid Rise	215
RES6	R6MB	USACE - Galveston	Nursing Home, structure	Mid Rise	215
RES6	R6HN	USACE - Galveston	Nursing Home, structure	High Rise	215
RES6	R6HB	USACE - Galveston	Nursing Home, structure	High Rise	215
COM1	C1LN	USACE - Galveston	Average Retail, Structure	Low Rise	217
COM1	C1LB	USACE - Galveston	Average Retail, Structure	Low Rise	217
COM1	C1MN	USACE - Galveston	Average Retail, Structure	Mid Rise	217

Occupancy	SpecificOccupId	Source	Description	Stories	ID
COM1	C1MB	USACE - Galveston	Average Retail, Structure	Mid Rise	217
COM1	C1HN	USACE - Galveston	Average Retail, Structure	High Rise	217
COM1	C1HB	USACE - Galveston	Average Retail, Structure	High Rise	217
COM2	C2LN	USACE - Galveston	Average wholesale, Structure	Low Rise	341
COM2	C2LB	USACE - Galveston	Average wholesale, Structure	Low Rise	341
COM2	C2MN	USACE - Galveston	Average wholesale, Structure	Mid Rise	341
COM2	C2MB	USACE - Galveston	Average wholesale, Structure	Mid Rise	341
COM2	C2HN	USACE - Galveston	Average wholesale, Structure	High Rise	341
COM2	C2HB	USACE - Galveston	Average wholesale, Structure	High Rise	341
COM3	C3LN	USACE - Galveston	Average Personal & Repair Services, Structure	Low Rise	375
COM3	C3LB	USACE - Galveston	Average Personal & Repair Services, Structure	Low Rise	375
COM3	C3MN	USACE - Galveston	Average Personal & Repair Services, Structure	Mid Rise	375
COM3	C3MB	USACE - Galveston	Average Personal & Repair Services, Structure	Mid Rise	375
COM3	C3HN	USACE - Galveston	Average Personal & Repair Services, Structure	High Rise	375
COM3	C3HB	USACE - Galveston	Average Personal & Repair Services, Structure	High Rise	375
COM4	C4LN	USACE - Galveston	Average Prof/Tech Services, Structure	Low Rise	431

Occupancy	SpecificOccupId	Source	Description	Stories	ID
COM4	C4LB	USACE - Galveston	Average Prof/Tech Services, Structure	Low Rise	431
COM4	C4MN	USACE - Galveston	Average Prof/Tech Services, Structure	Mid Rise	431
COM4	C4MB	USACE - Galveston	Average Prof/Tech Services, Structure	Mid Rise	431
COM4	C4HN	USACE - Galveston	Average Prof/Tech Services, Structure	High Rise	431
COM4	C4HB	USACE - Galveston	Average Prof/Tech Services, Structure	High Rise	431
COM5	C5LN	USACE - Galveston	Bank, Structure	Low Rise	467
COM5	C5LB	USACE - Galveston	Bank, Structure	Low Rise	467
COM5	C5MN	USACE - Galveston	Bank, Structure	Mid Rise	467
COM5	C5MB	USACE - Galveston	Bank, Structure	Mid Rise	467
COM5	C5HN	USACE - Galveston	Bank, Structure	High Rise	467
COM5	C5HB	USACE - Galveston	Bank, Structure	High Rise	467
COM6	C6LN	USACE - Galveston	Hospital, Structure	Low Rise	474
COM6	C6LB	USACE - Galveston	Hospital, Structure	Low Rise	474
COM6	C6MN	USACE - Galveston	Hospital, Structure	Mid Rise	474
COM6	C6MB	USACE - Galveston	Hospital, Structure	Mid Rise	474
COM6	C6HN	USACE - Galveston	Hospital, Structure	High Rise	474

Occupancy	SpecificOccupId	Source	Description	Stories	ID
COM6	C6HB	USACE - Galveston	Hospital, Structure	High Rise	474
COM7	C7LN	USACE - Galveston	Average Medical Office, Structure	Low Rise	475
COM7	C7LB	USACE - Galveston	Average Medical Office, Structure	Low Rise	475
COM7	C7MN	USACE - Galveston	Average Medical Office, Structure	Mid Rise	475
COM7	C7MB	USACE - Galveston	Average Medical Office, Structure	Mid Rise	475
COM7	C7HN	USACE - Galveston	Average Medical Office, Structure	High Rise	475
COM7	C7HB	USACE - Galveston	Average Medical Office, Structure	High Rise	475
COM8	C8LN	USACE - Galveston	Average Entertainment/Recreation, Structure	Low Rise	493
COM8	C8LB	USACE - Galveston	Average Entertainment/Recreation, Structure	Low Rise	493
COM8	C8MN	USACE - Galveston	Average Entertainment/Recreation, Structure	Mid Rise	493
COM8	C8MB	USACE - Galveston	Average Entertainment/Recreation, Structure	Mid Rise	493
COM8	C8HN	USACE - Galveston	Average Entertainment/Recreation, Structure	High Rise	493
COM8	C8HB	USACE - Galveston	Average Entertainment/Recreation, Structure	High Rise	493
COM9	C9LN	USACE - Galveston	Average theatre, Structure	Low Rise	532
COM9	C9LB	USACE - Galveston	Average theatre, Structure	Low Rise	532
COM9	C9MN	USACE - Galveston	Average theatre, Structure	Mid Rise	532

Occupancy	SpecificOccupId	Source	Description	Stories	ID
COM9	C9MB	USACE - Galveston	Average theatre, Structure	Mid Rise	532
COM9	C9HN	USACE - Galveston	Average theatre, Structure	High Rise	532
COM9	C9HB	USACE - Galveston	Average theatre, Structure	High Rise	532
COM10	C10LN	USACE - Galveston	Garage, structure	Low Rise	543
COM10	C10LB	USACE - Galveston	Garage, structure	Low Rise	543
COM10	C10MN	USACE - Galveston	Garage, structure	Mid Rise	543
COM10	C10MB	USACE - Galveston	Garage, structure	Mid Rise	543
COM10	C10HN	USACE - Galveston	Garage, structure	High Rise	543
COM10	C10HB	USACE - Galveston	Garage, structure	High Rise	543
IND1	I1LN	USACE - Galveston	Average heavy industrial, Structure	Low Rise	545
IND1	I1LB	USACE - Galveston	Average heavy industrial, Structure	Low Rise	545
IND1	I1MN	USACE - Galveston	Average heavy industrial, Structure	Mid Rise	545
IND1	I1MB	USACE - Galveston	Average heavy industrial, Structure	Mid Rise	545
IND1	I1HN	USACE - Galveston	Average heavy industrial, Structure	High Rise	545
IND1	I1HB	USACE - Galveston	Average heavy industrial, Structure	High Rise	545
IND2	I2LN	USACE - Galveston	Average light industrial, structure	Low Rise	559

Occupancy	SpecificOccupId	Source	Description	Stories	ID
IND2	I2LB	USACE - Galveston	Average light industrial, structure	Low Rise	559
IND2	I2MN	USACE - Galveston	Average light industrial, structure	Mid Rise	559
IND2	I2MB	USACE - Galveston	Average light industrial, structure	Mid Rise	559
IND2	I2HN	USACE - Galveston	Average light industrial, structure	High Rise	559
IND2	I2HB	USACE - Galveston	Average light industrial, structure	High Rise	559
IND3	I3LN	USACE - Galveston	Average Food/Drug/Chem, Structure	Low Rise	575
IND3	I3LB	USACE - Galveston	Average Food/Drug/Chem, Structure	Low Rise	575
IND3	I3MN	USACE - Galveston	Average Food/Drug/Chem, Structure	Mid Rise	575
IND3	I3MB	USACE - Galveston	Average Food/Drug/Chem, Structure	Mid Rise	575
IND3	I3HN	USACE - Galveston	Average Food/Drug/Chem, Structure	High Rise	575
IND3	I3HB	USACE - Galveston	Average Food/Drug/Chem, Structure	High Rise	575
IND4	I4LN	USACE - Galveston	Average Metals/Minerals processing, structure	Low Rise	586
IND4	I4LB	USACE - Galveston	Average Metals/Minerals processing, structure	Low Rise	586
IND4	I4MN	USACE - Galveston	Average Metals/Minerals processing, structure	Mid Rise	586
IND4	I4MB	USACE - Galveston	Average Metals/Minerals processing, structure	Mid Rise	586
IND4	I4HN	USACE - Galveston	Average Metals/Minerals processing, structure	High Rise	586

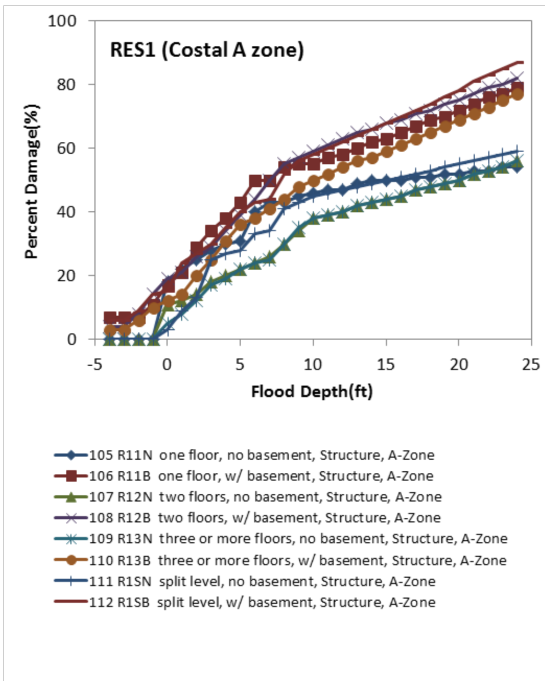
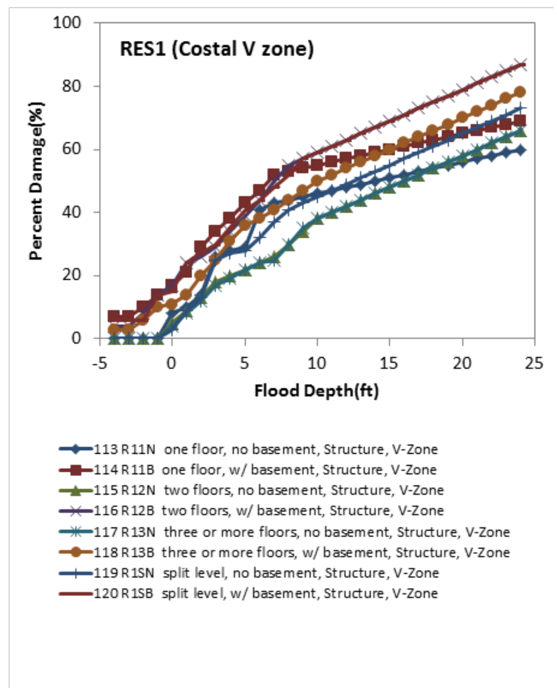
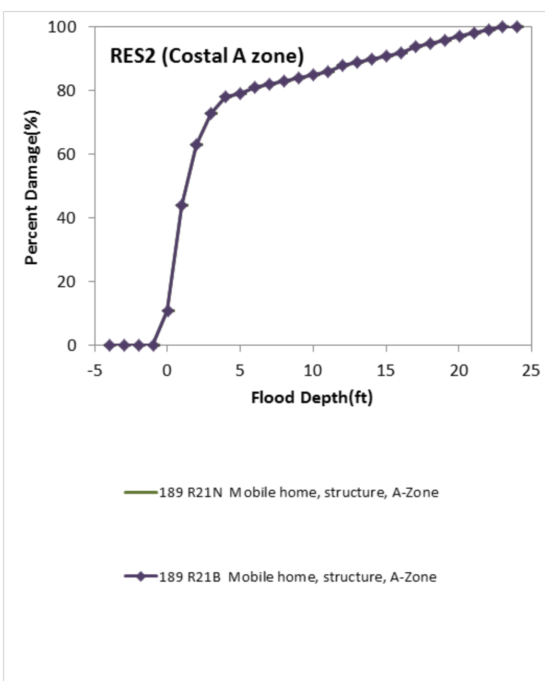
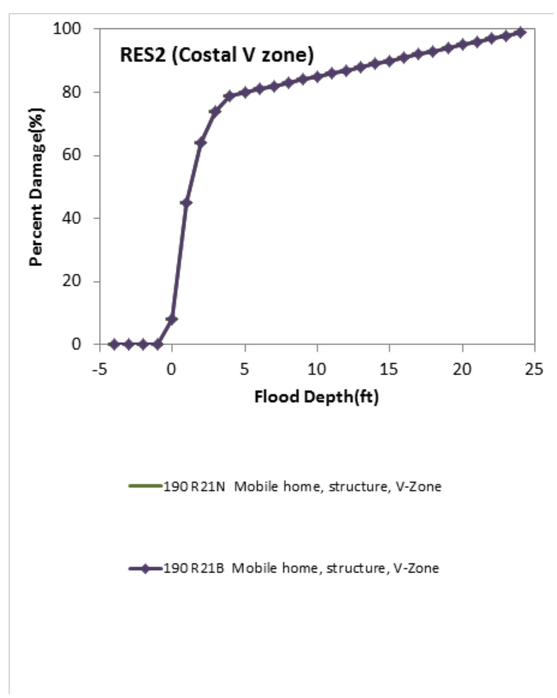
Occupancy	SpecificOccupId	Source	Description	Stories	ID
IND4	I4HB	USACE - Galveston	Average Metals/Minerals processing, structure	High Rise	586
IND5	I5LN	USACE - Galveston	Average High Technology, structure	Low Rise	591
IND5	I5LB	USACE - Galveston	Average High Technology, structure	Low Rise	591
IND5	I5MN	USACE - Galveston	Average High Technology, structure	Mid Rise	591
IND5	I5MB	USACE - Galveston	Average High Technology, structure	Mid Rise	591
IND5	I5HN	USACE - Galveston	Average High Technology, structure	High Rise	591
IND5	I5HB	USACE - Galveston	Average High Technology, structure	High Rise	591
IND6	I6LN	USACE - Galveston	Average Construction, structure	Low Rise	592
IND6	I6LB	USACE - Galveston	Average Construction, structure	Low Rise	592
IND6	I6MN	USACE - Galveston	Average Construction, structure	Mid Rise	592
IND6	I6MB	USACE - Galveston	Average Construction, structure	Mid Rise	592
IND6	I6HN	USACE - Galveston	Average Construction, structure	High Rise	592
IND6	I6HB	USACE - Galveston	Average Construction, structure	High Rise	592
AGR1	A1LN	USACE - Galveston	Average Agriculture, structure	Low Rise	616
AGR1	A1LB	USACE - Galveston	Average Agriculture, structure	Low Rise	616
AGR1	A1MN	USACE - Galveston	Average Agriculture, structure	Mid Rise	616

Occupancy	SpecificOccupId	Source	Description	Stories	ID
AGR1	A1MB	USACE - Galveston	Average Agriculture, structure	Mid Rise	616
AGR1	A1HN	USACE - Galveston	Average Agriculture, structure	High Rise	616
AGR1	A1HB	USACE - Galveston	Average Agriculture, structure	High Rise	616
REL1	RE1LN	USACE - Galveston	Church, structure	Low Rise	624
REL1	RE1LB	USACE - Galveston	Church, structure	Low Rise	624
REL1	RE1MN	USACE - Galveston	Church, structure	Mid Rise	624
REL1	RE1MB	USACE - Galveston	Church, structure	Mid Rise	624
REL1	RE1HN	USACE - Galveston	Church, structure	High Rise	624
REL1	RE1HB	USACE - Galveston	Church, structure	High Rise	624
GOV1	G1LN	USACE - Galveston	Average government services, structure	Low Rise	631
GOV1	G1LB	USACE - Galveston	Average government services, structure	Low Rise	631
GOV1	G1MN	USACE - Galveston	Average government services, structure	Mid Rise	631
GOV1	G1MB	USACE - Galveston	Average government services, structure	Mid Rise	631
GOV1	G1HN	USACE - Galveston	Average government services, structure	High Rise	631
GOV1	G1HB	USACE - Galveston	Average government services, structure	High Rise	631
GOV2	G2LN	USACE - Galveston	Average emergency response, Structure	Low Rise	640

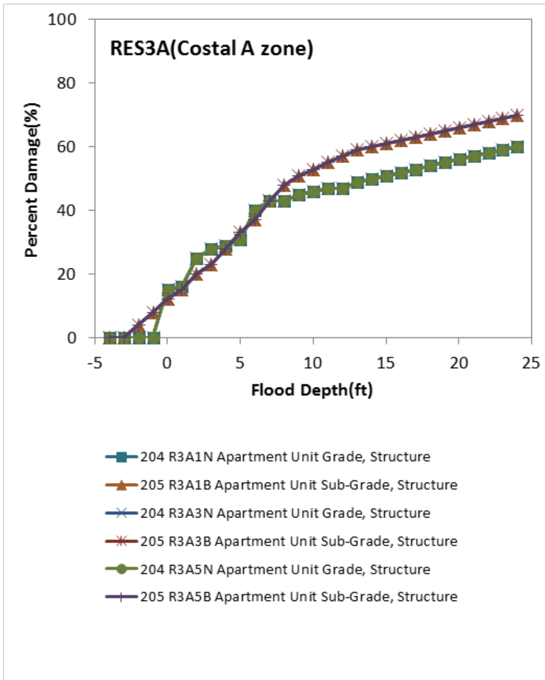
Occupancy	SpecificOccupId	Source	Description	Stories	ID
GOV2	G2LB	USACE - Galveston	Average emergency response, Structure	Low Rise	640
GOV2	G2MN	USACE - Galveston	Average emergency response, Structure	Mid Rise	640
GOV2	G2MB	USACE - Galveston	Average emergency response, Structure	Mid Rise	640
GOV2	G2HN	USACE - Galveston	Average emergency response, Structure	High Rise	640
GOV2	G2HB	USACE - Galveston	Average emergency response, Structure	High Rise	640
EDU1	E1LN	USACE - Galveston	Average school, structure	Low Rise	643
EDU1	E1LB	USACE - Galveston	Average school, structure	Low Rise	643
EDU1	E1MN	USACE - Galveston	Average school, structure	Mid Rise	643
EDU1	E1MB	USACE - Galveston	Average school, structure	Mid Rise	643
EDU1	E1HN	USACE - Galveston	Average school, structure	High Rise	643
EDU1	E1HB	USACE - Galveston	Average school, structure	High Rise	643
EDU2	E2LN	USACE - Galveston	Average college/university, structure	Low Rise	652
EDU2	E2LB	USACE - Galveston	Average college/university, structure	Low Rise	652
EDU2	E2MN	USACE - Galveston	Average college/university, structure	Mid Rise	652
EDU2	E2MB	USACE - Galveston	Average college/university, structure	Mid Rise	652
EDU2	E2HN	USACE - Galveston	Average college/university, structure	High Rise	652

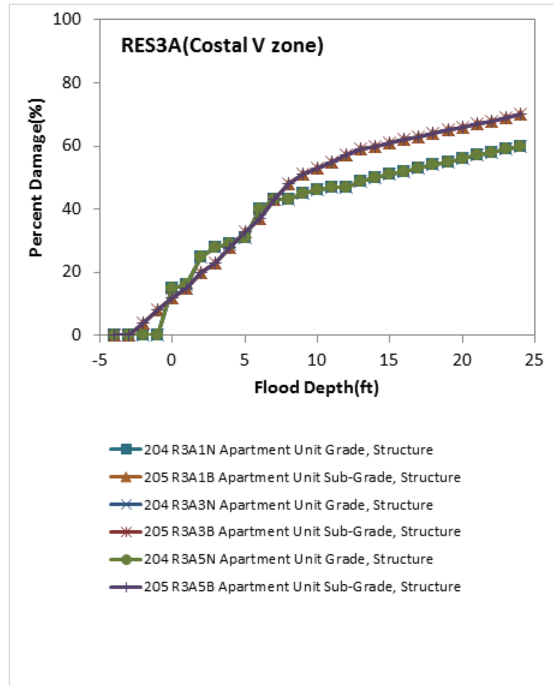
Occupancy	SpecificOccupId	Source	Description	Stories	ID
EDU2	E2HB	USACE - Galveston	Average college/university, structure	High Rise	652

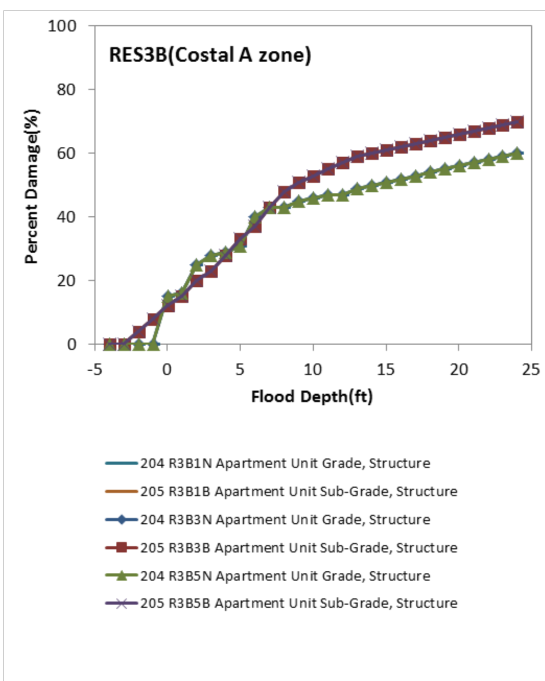
APPENDIX C HAZUS DAMAGE CURVES (A ZONE AND V ZONE)

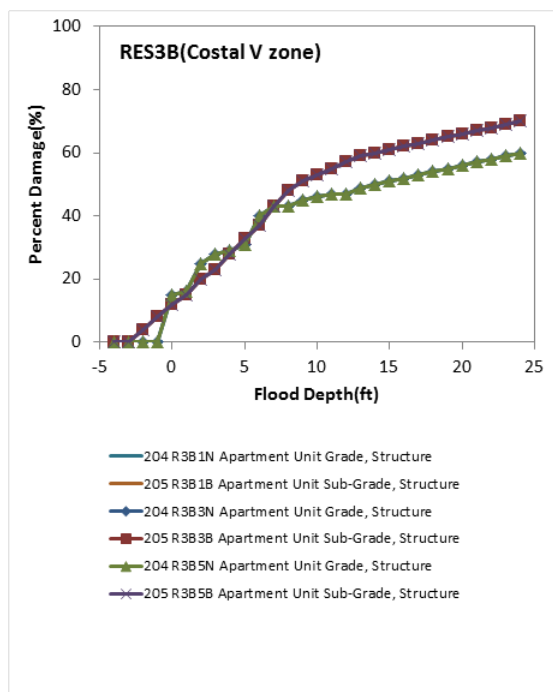
A zone curvs	V zone curves
	
	

A zone curvs	V zone curves

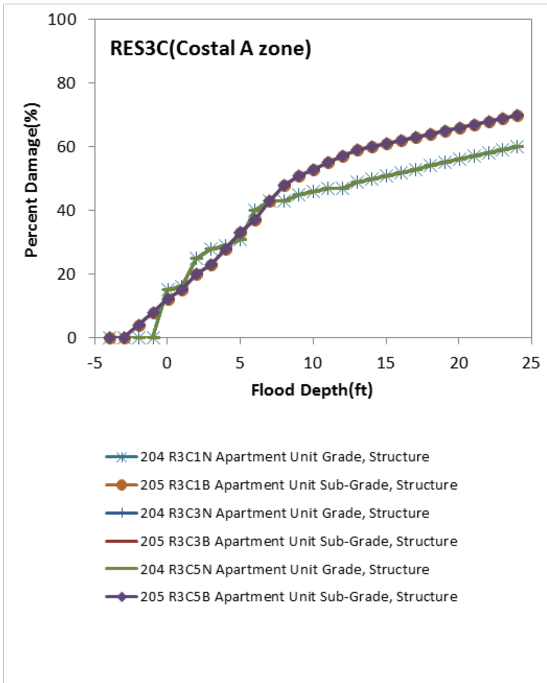


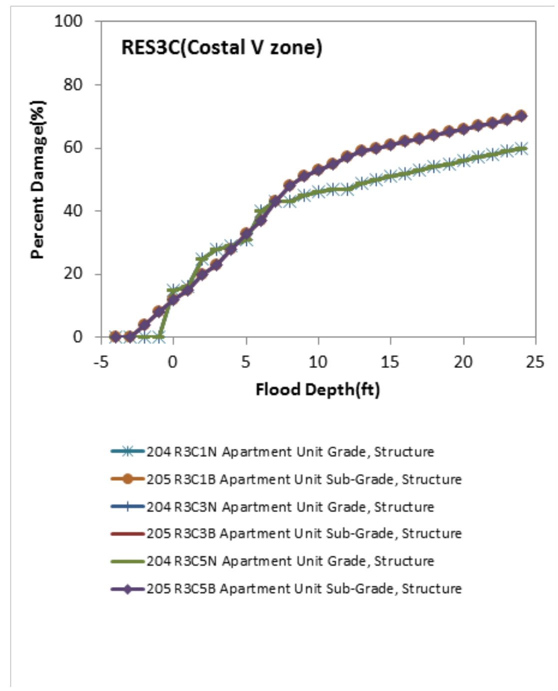


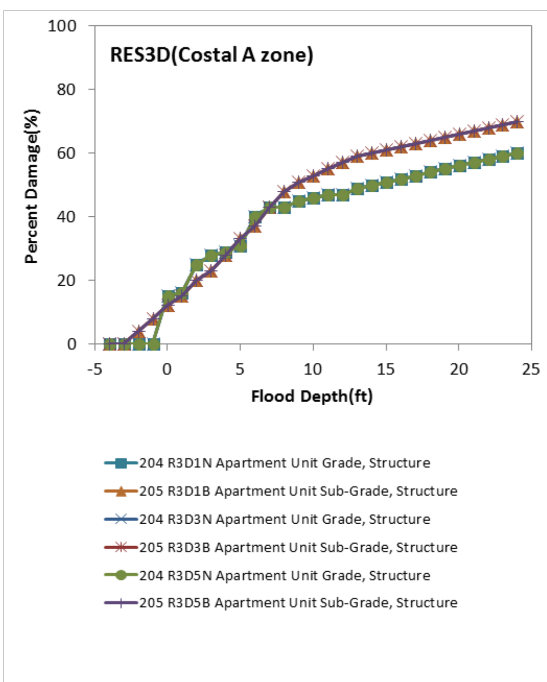


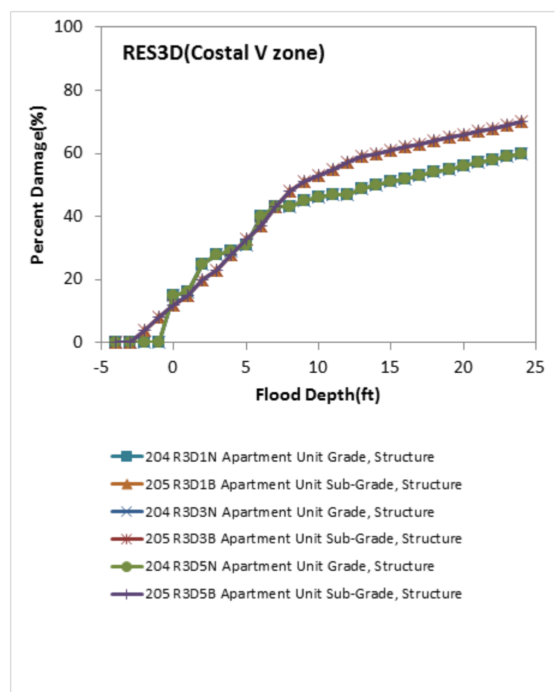


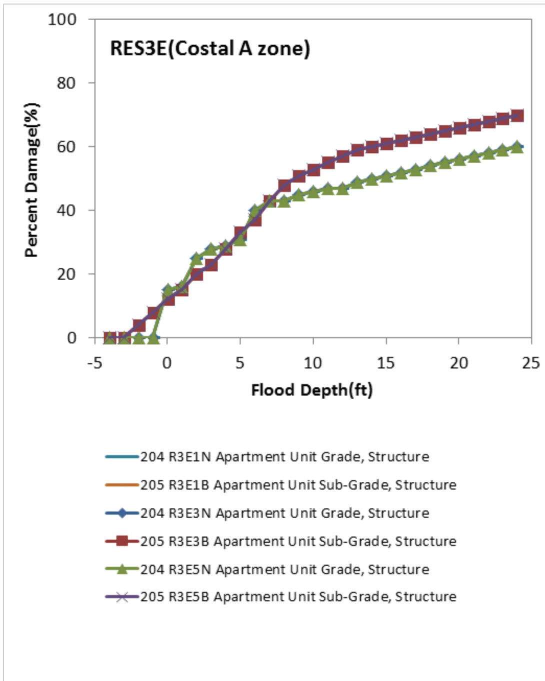
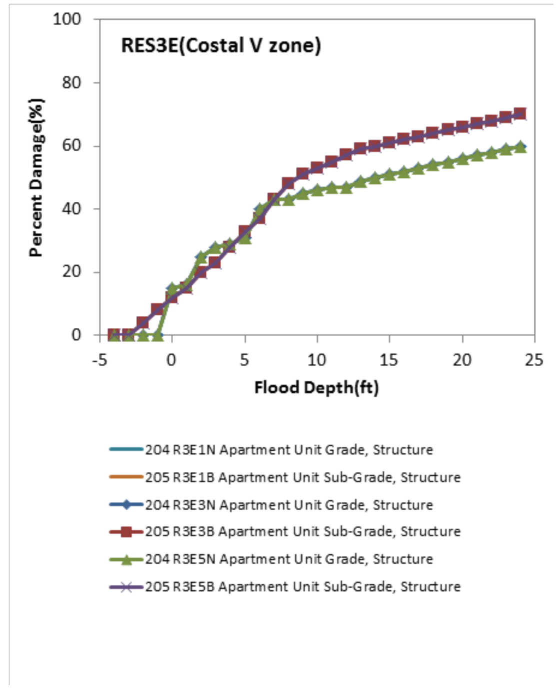
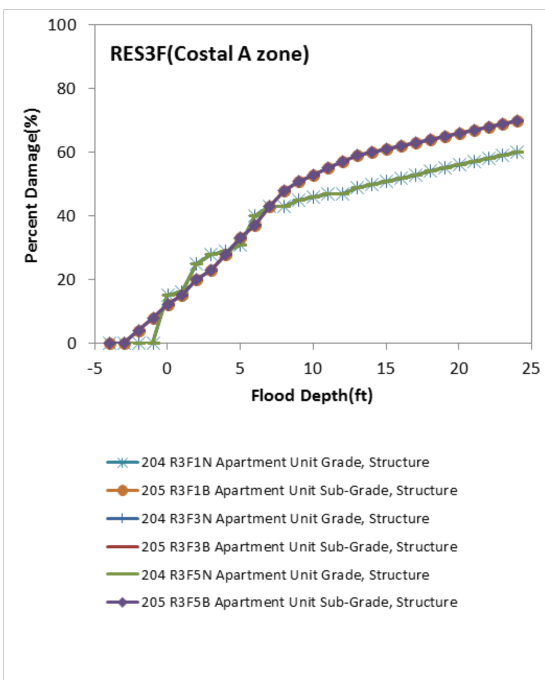
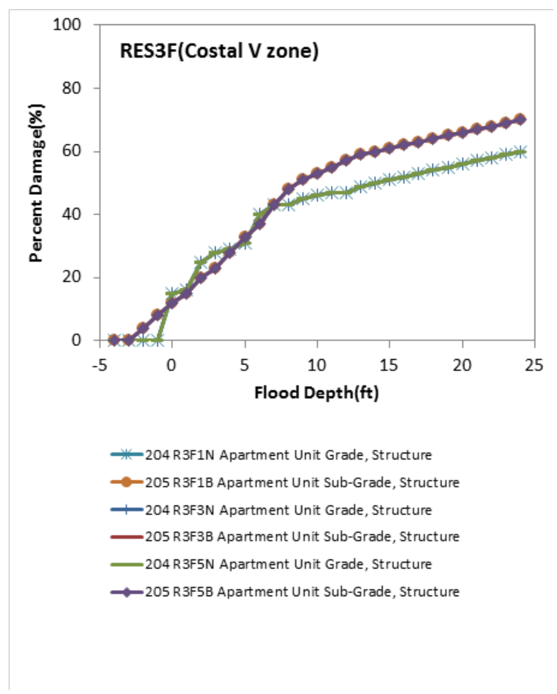
A zone curvs	V zone curves

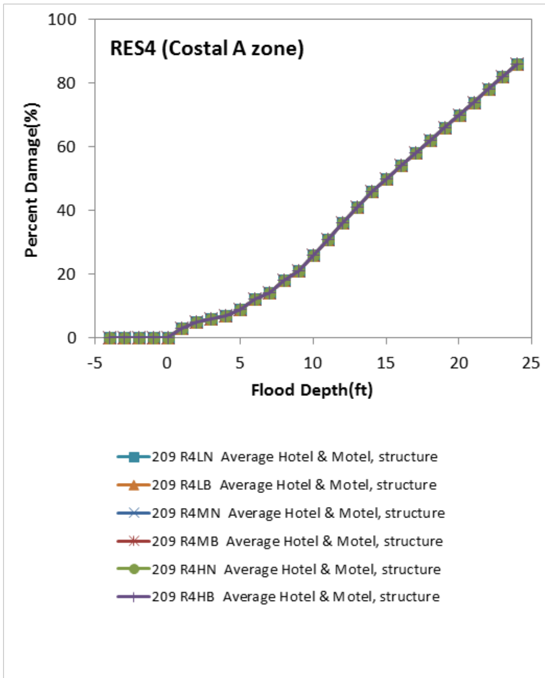


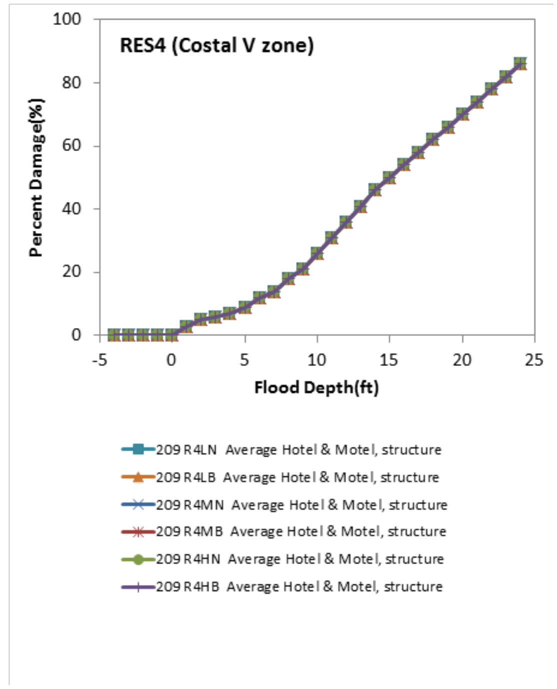


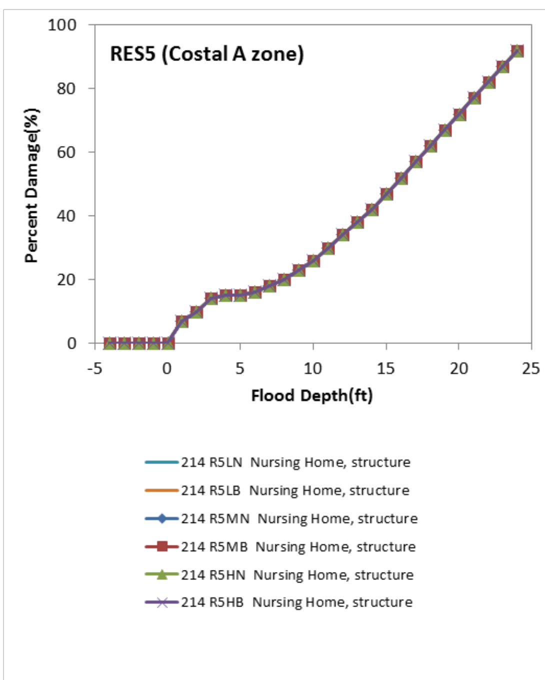


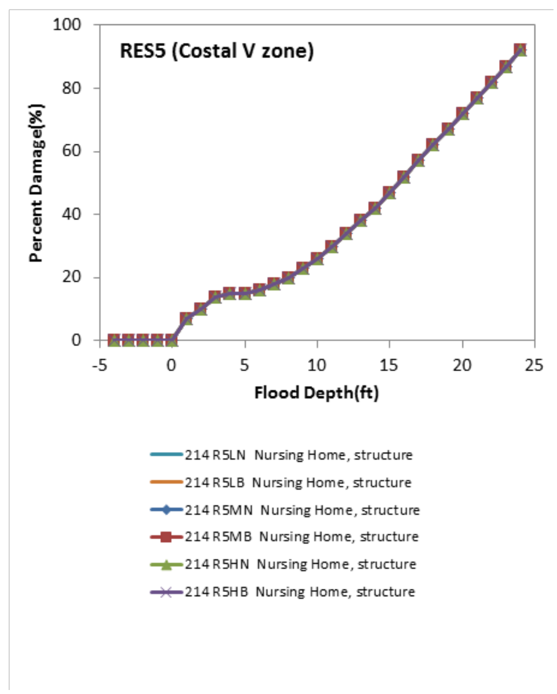


A zone curvs	V zone curves
	
	

<table>
<tr><th>A zone curvs</th><th>V zone curves</th></tr>
<tr><td>


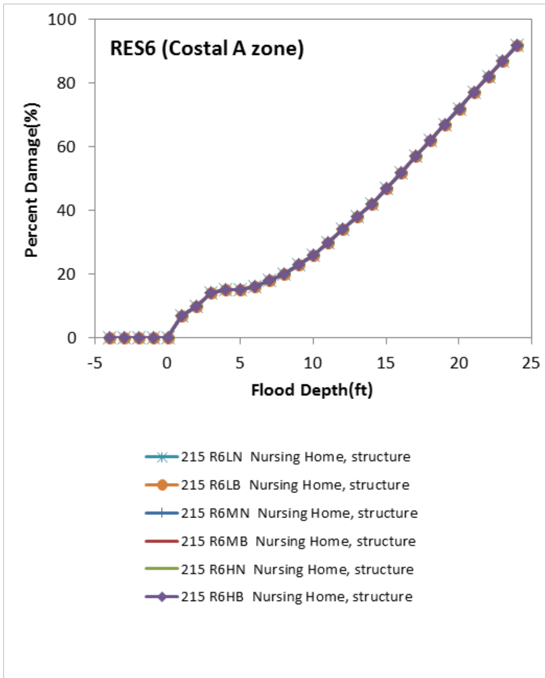
</td><td>


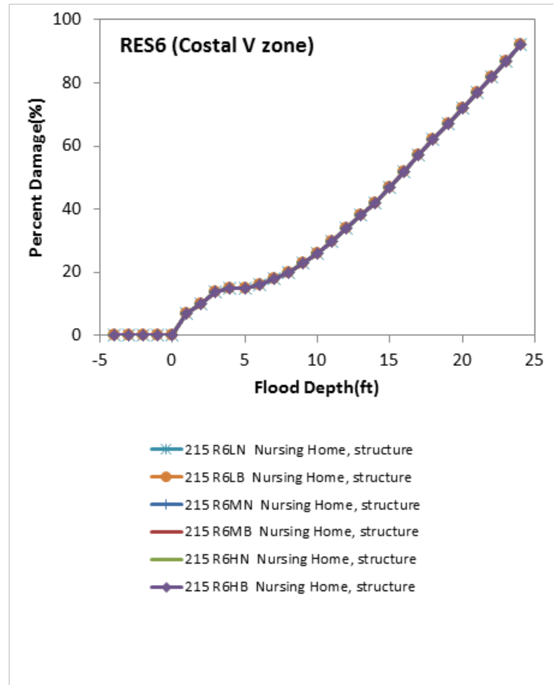
</td></tr>
<tr><td>


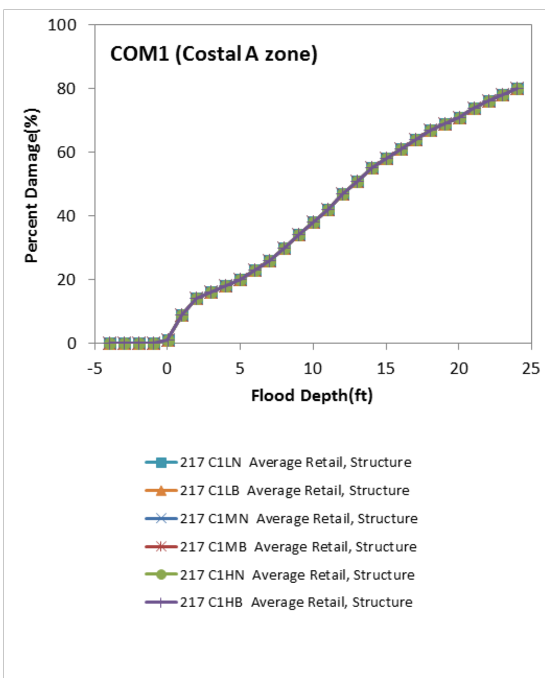
</td><td>


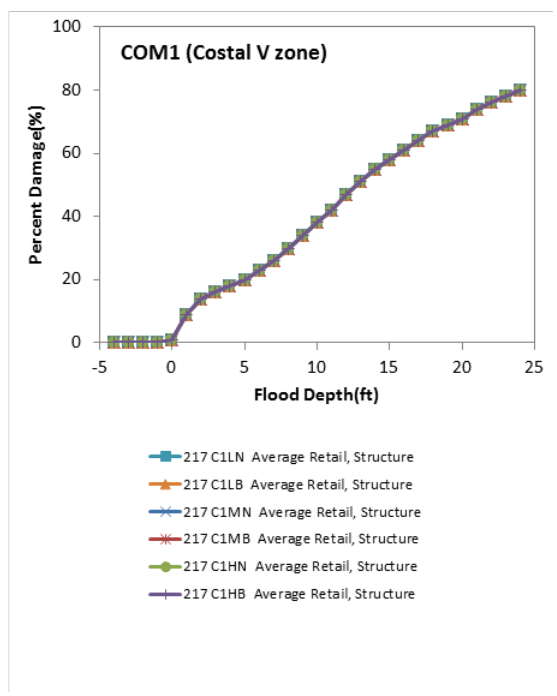
</td></tr>
</table>

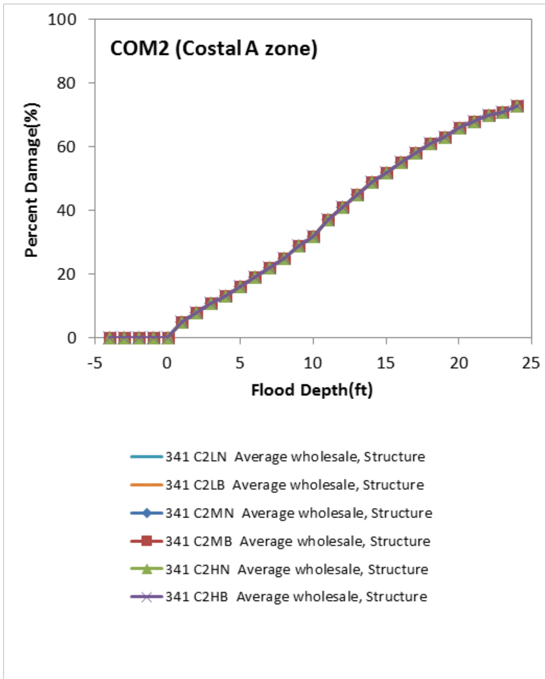
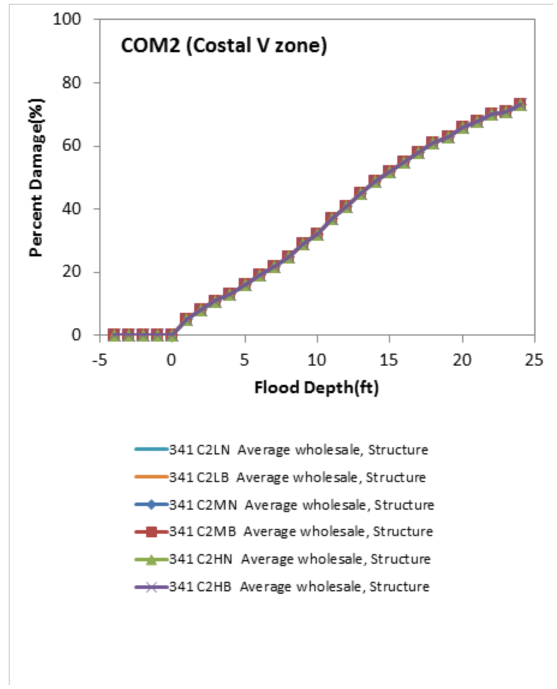
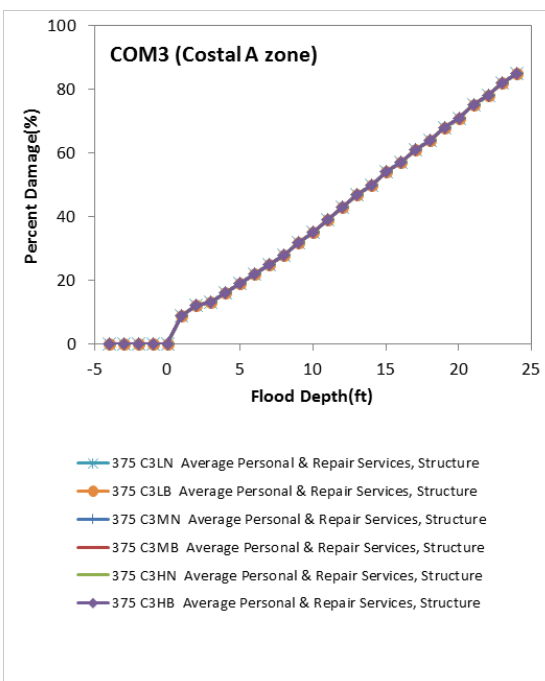
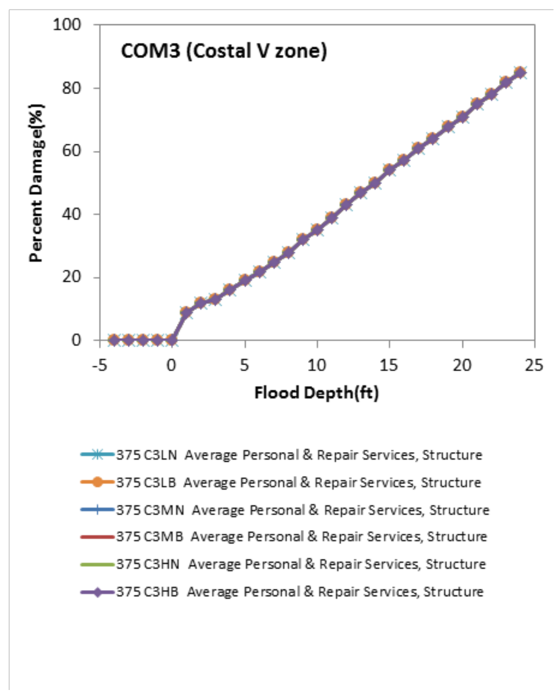
A zone curvs	V zone curves

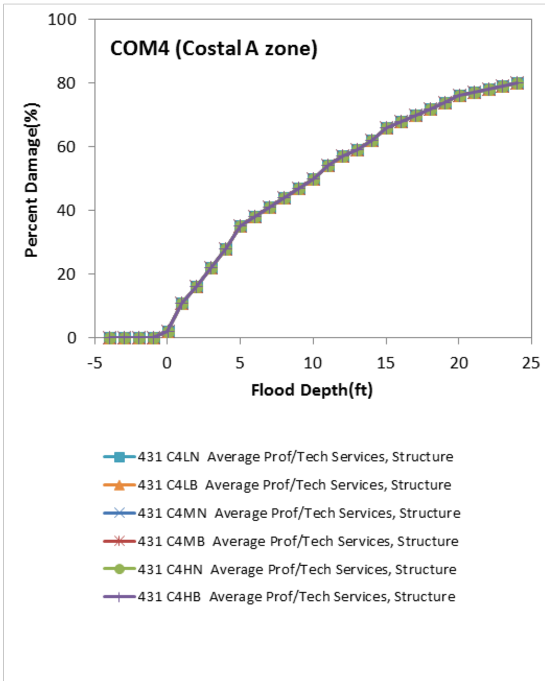
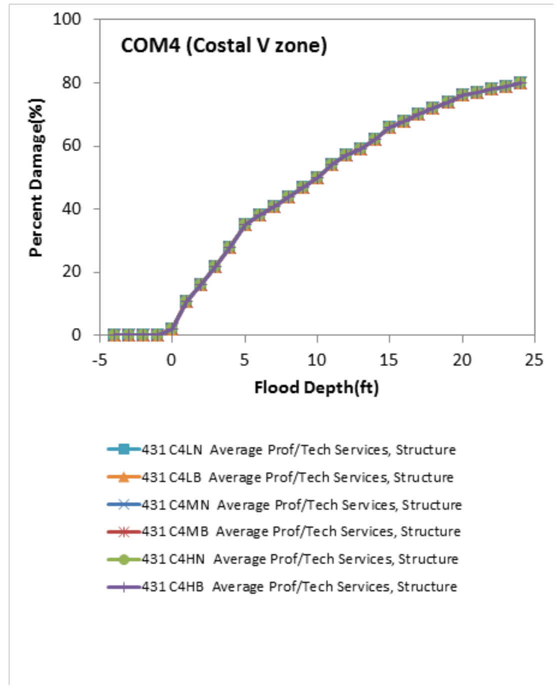
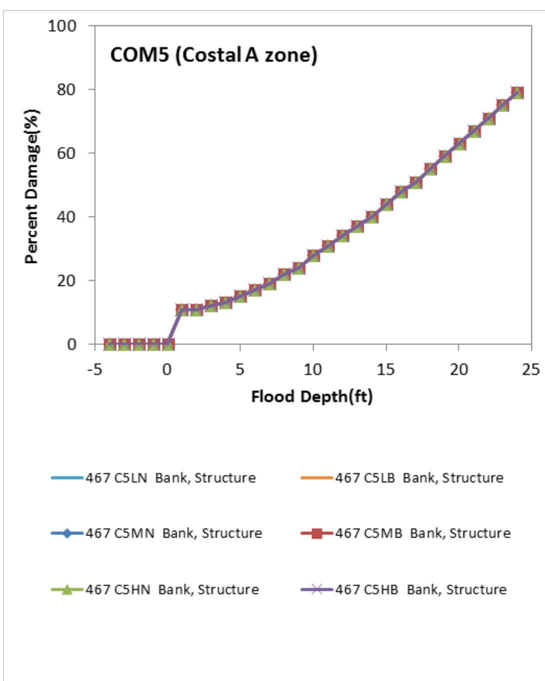
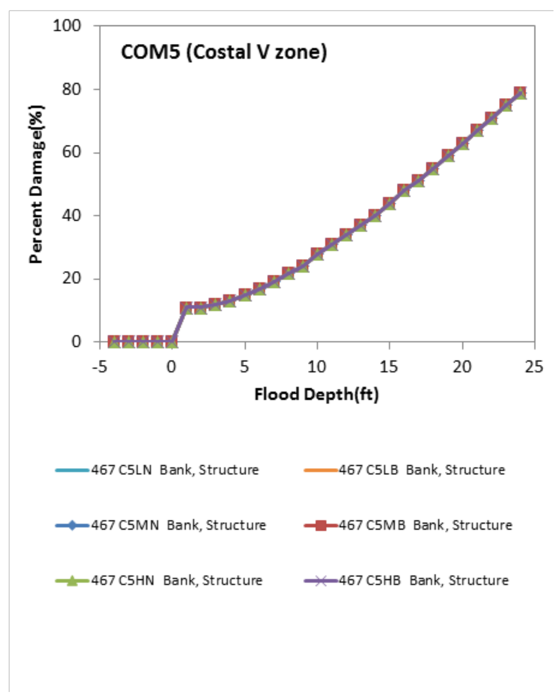


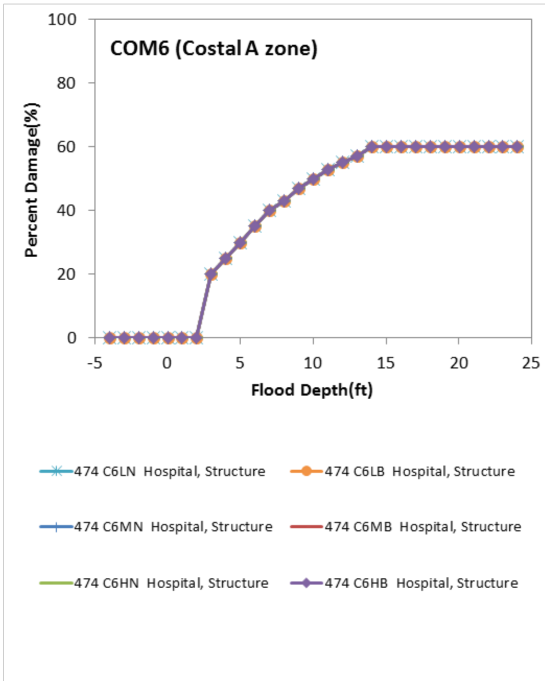


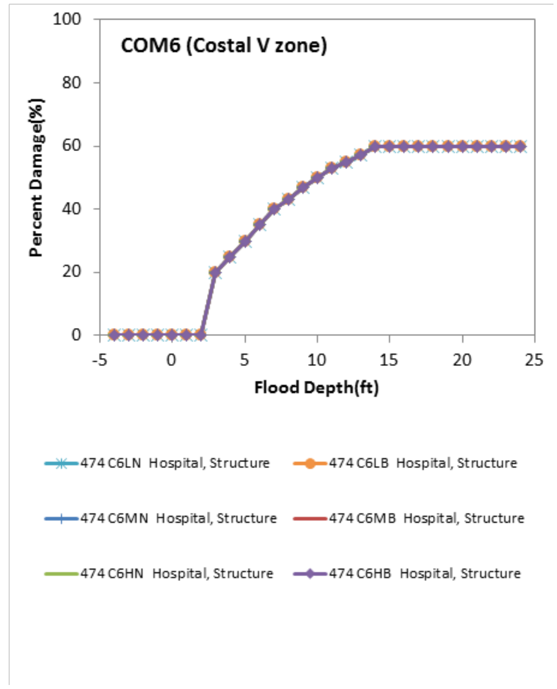


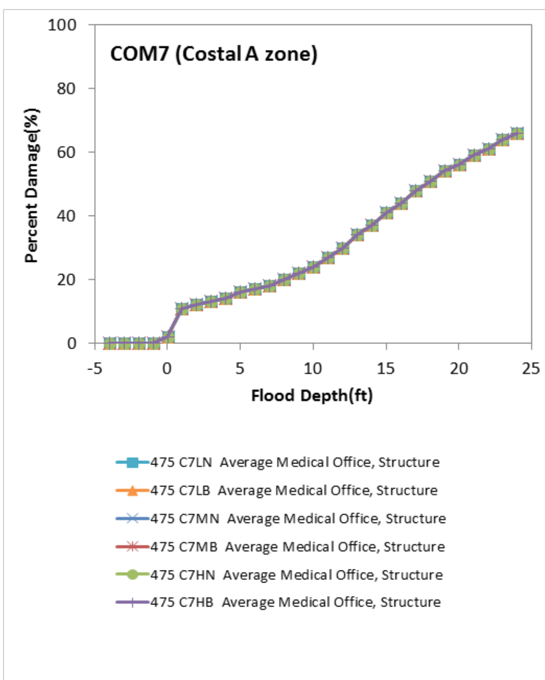


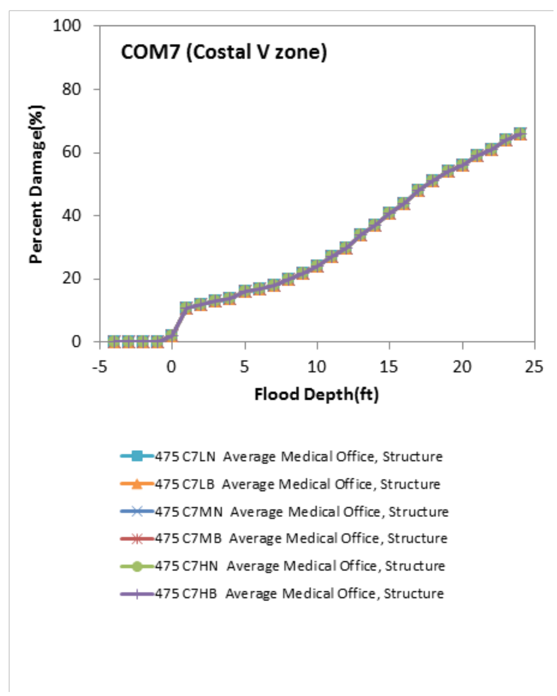
<table>
<tr><th>A zone curvs</th><th>V zone curves</th></tr>
<tr><td>
</td>
<td>
</td></tr>
<tr><td>
</td>
<td>
</td></tr>
</table>

<table>
<tr><th>A zone curvs</th><th>V zone curves</th></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>
</td><td>
</td></tr>
</table>

<table>
<tr><th>A zone curvs</th><th>V zone curves</th></tr>
<tr><td>


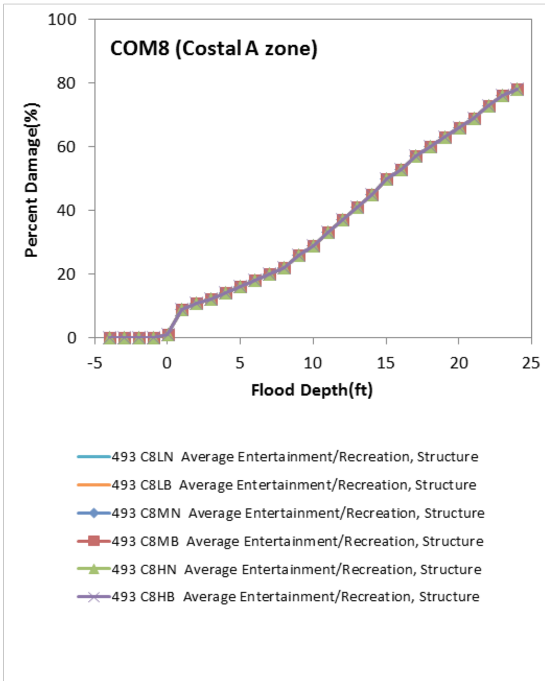
</td><td>


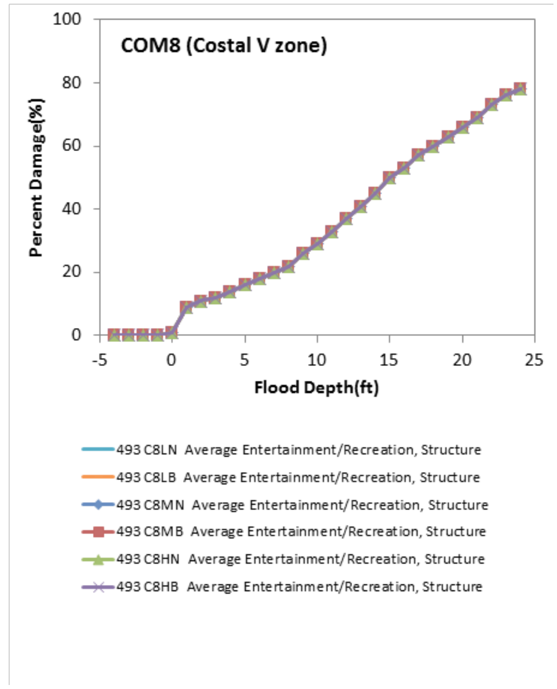
</td></tr>
<tr><td>


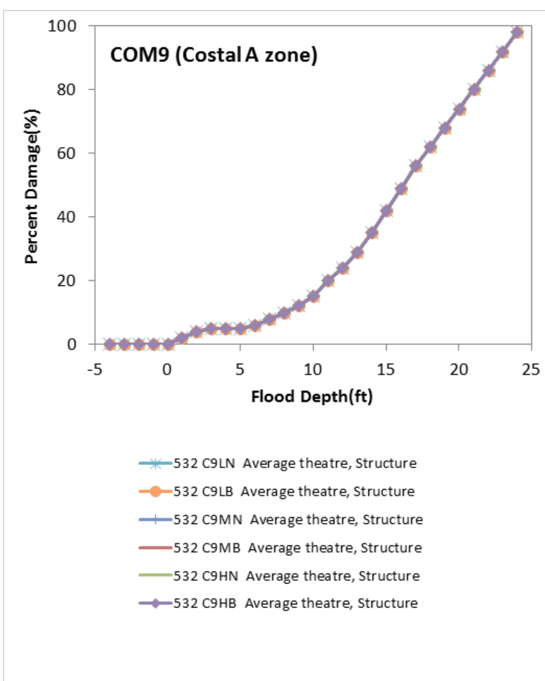
</td><td>


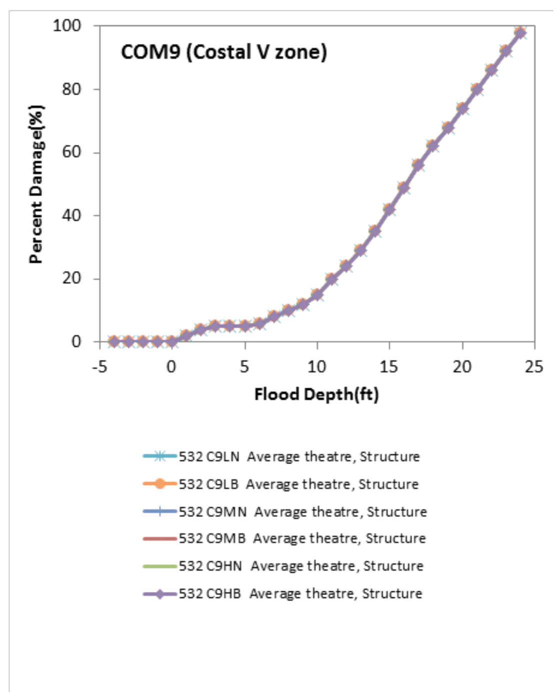
</td></tr>
</table>

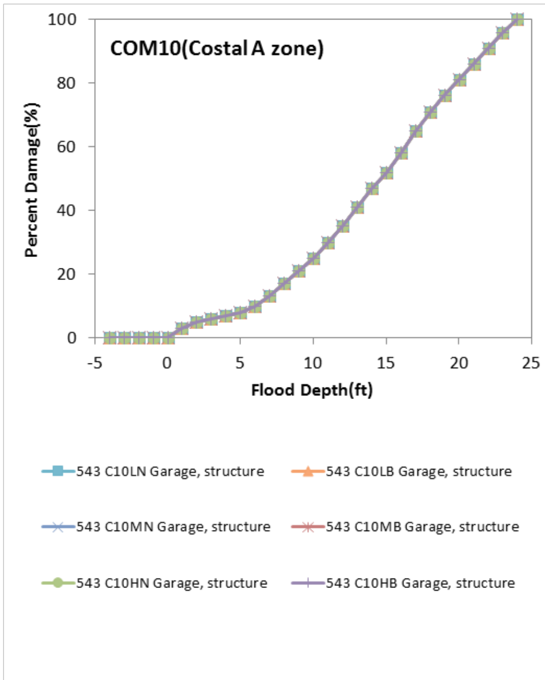
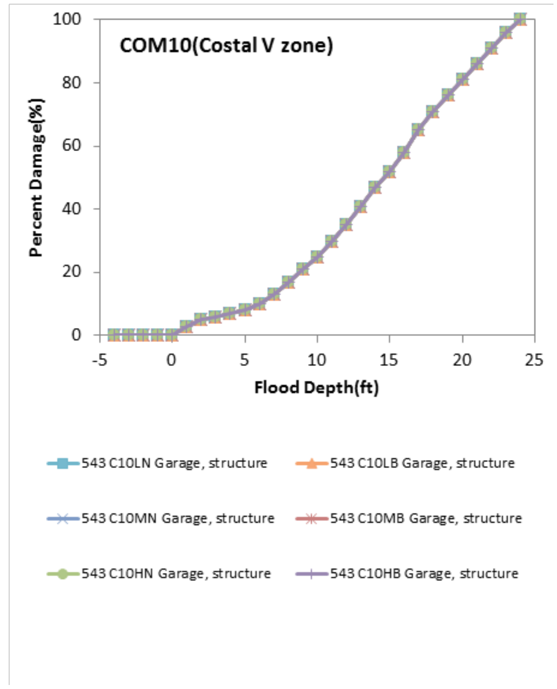
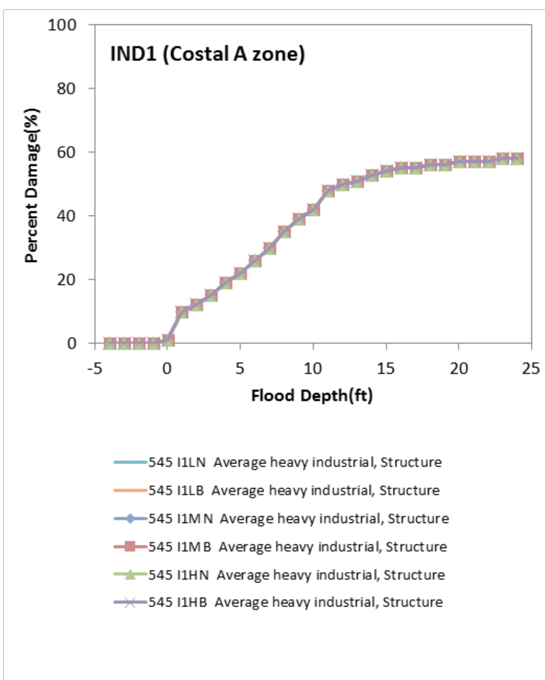
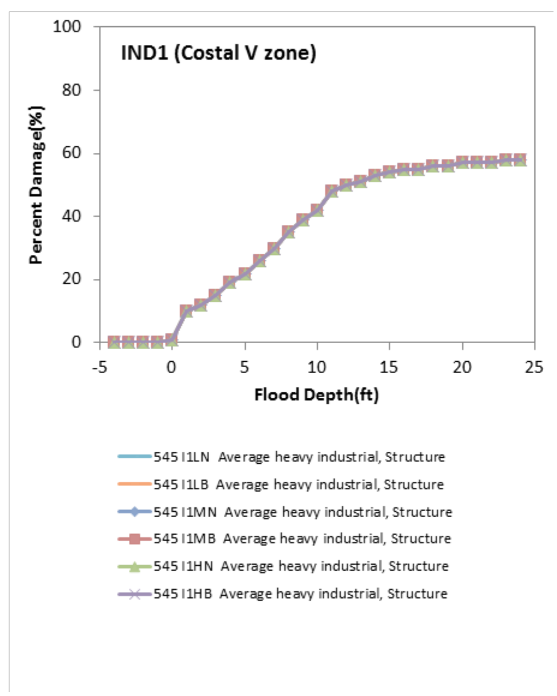
A zone curvs	V zone curves

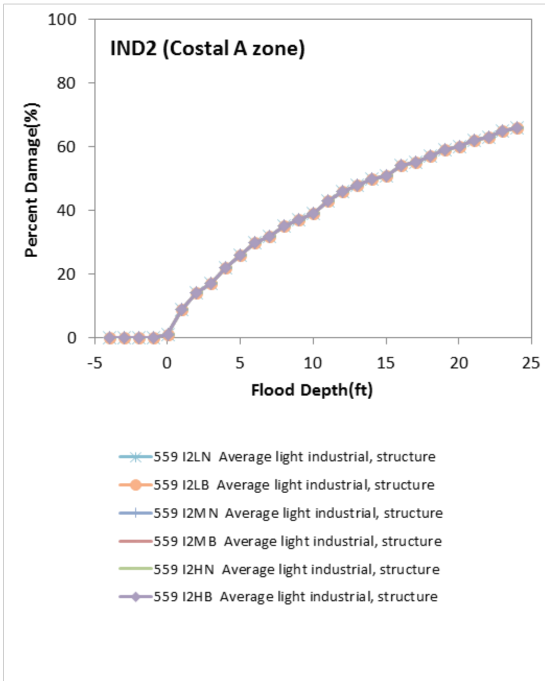
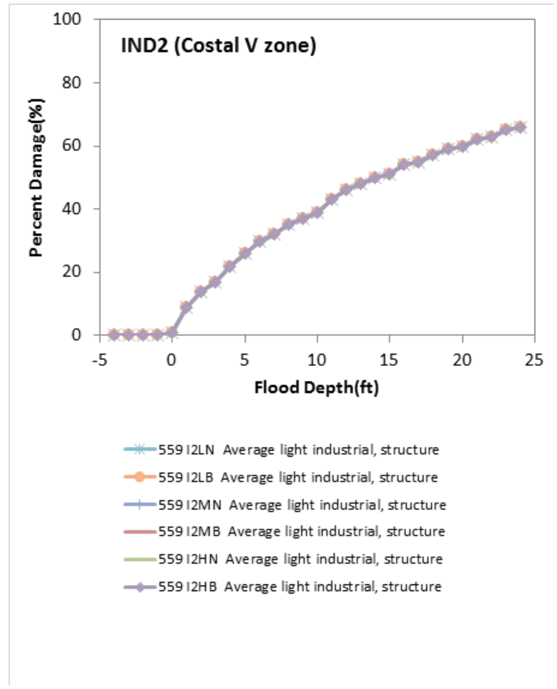
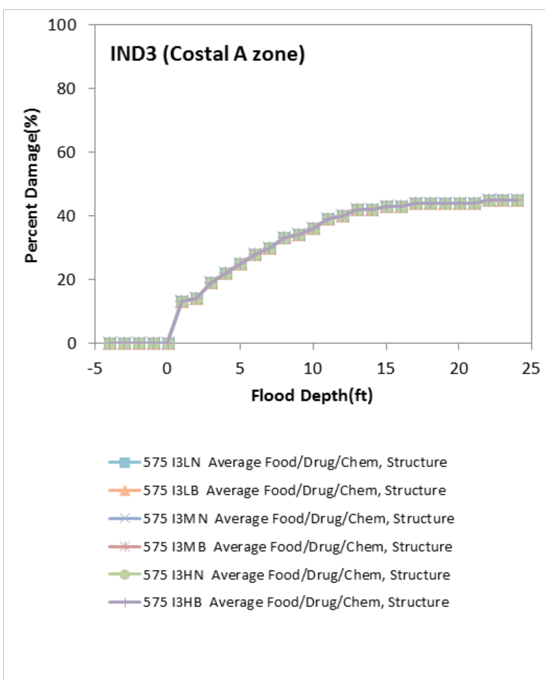
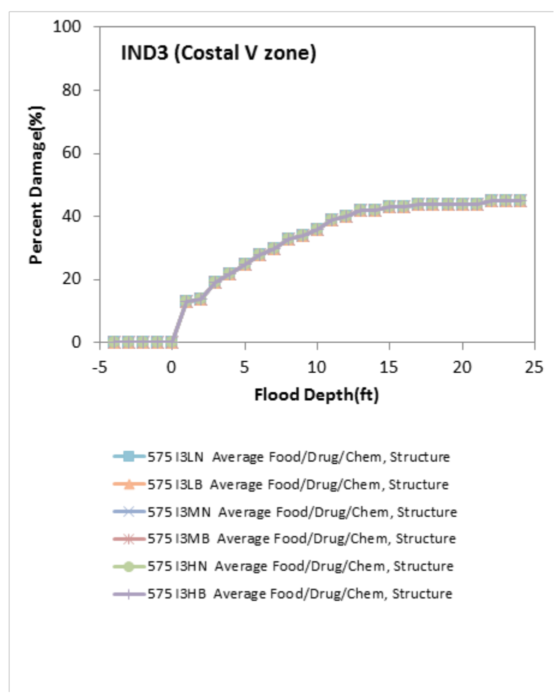


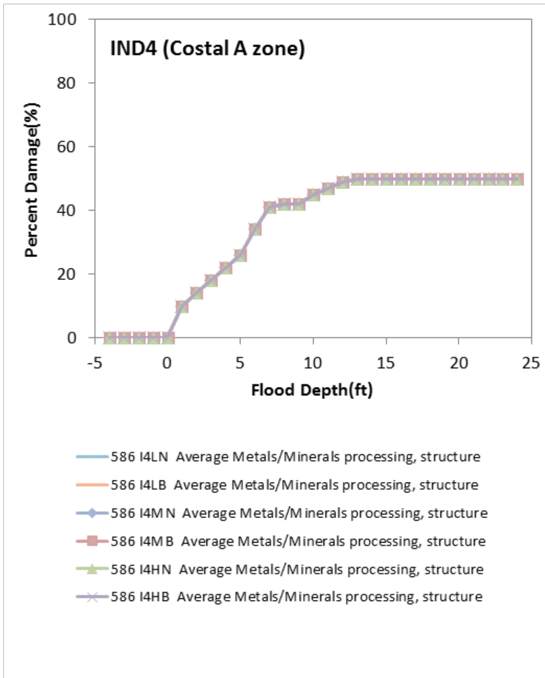


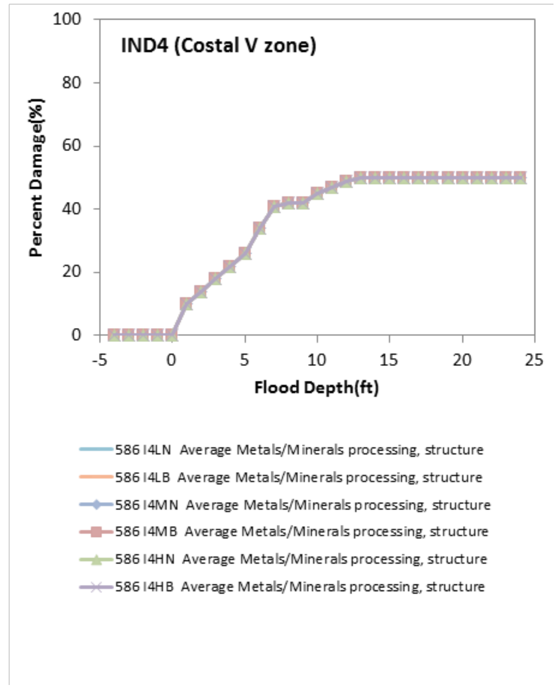


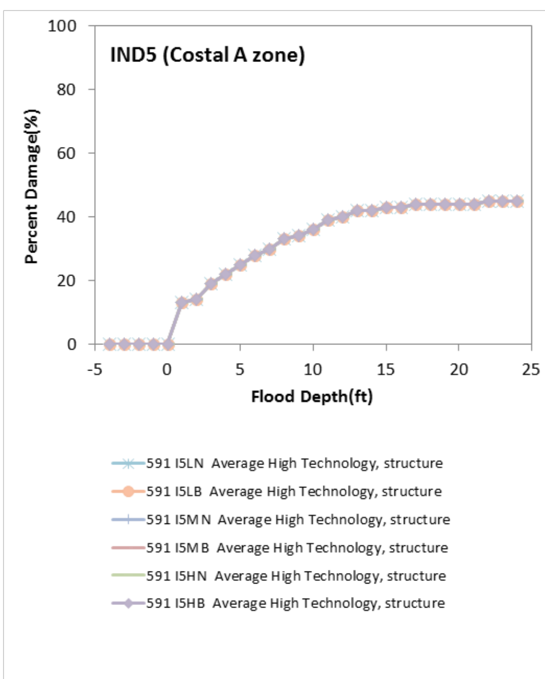


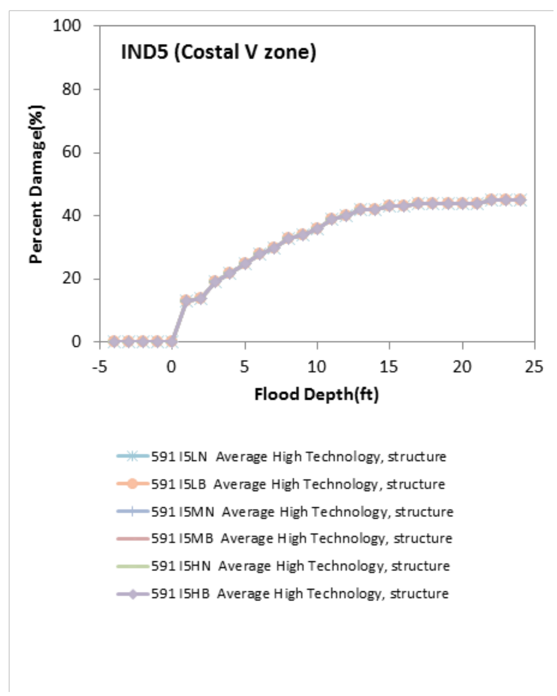
A zone curvs	V zone curves
	
	

<table>
<tr><th>A zone curvs</th><th>V zone curves</th></tr>
<tr><td>
</td>
<td>
</td></tr>
<tr><td>
</td>
<td>
</td></tr>
</table>

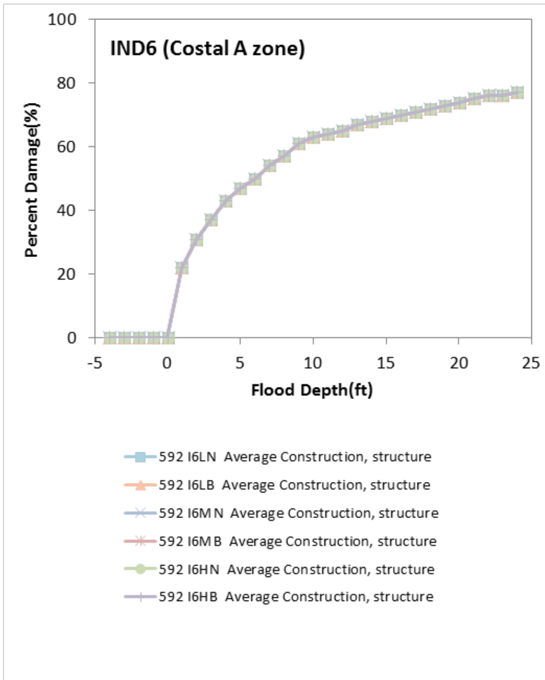
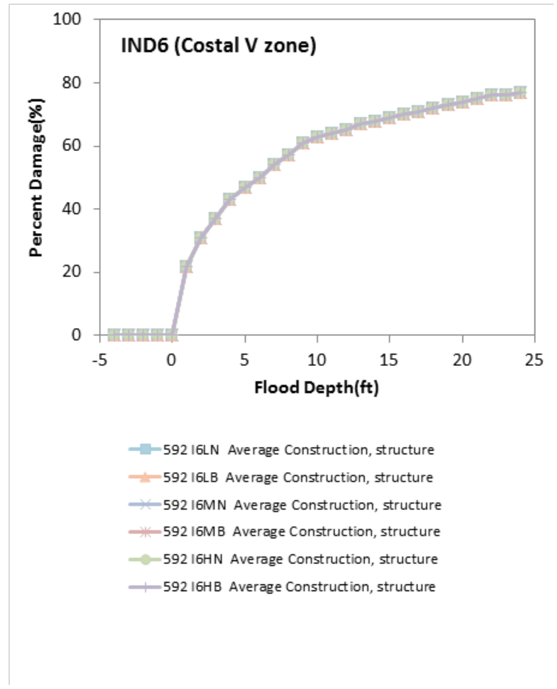
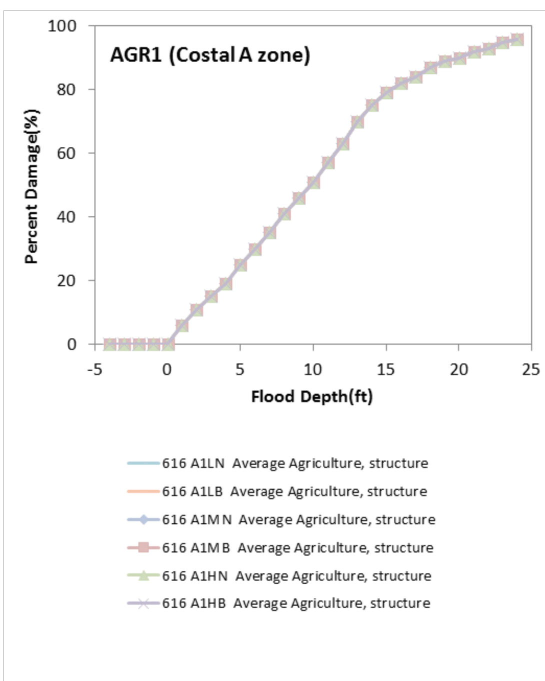
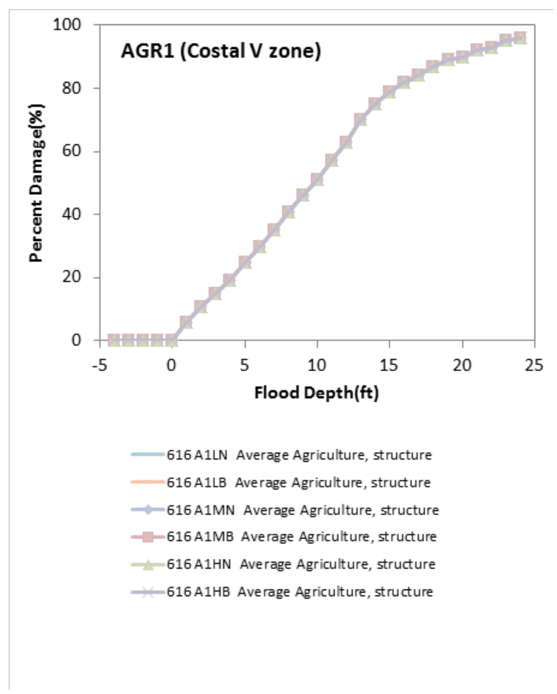
<table>
<tr><th>A zone curvs</th><th>V zone curves</th></tr>
<tr><td>


</td><td>


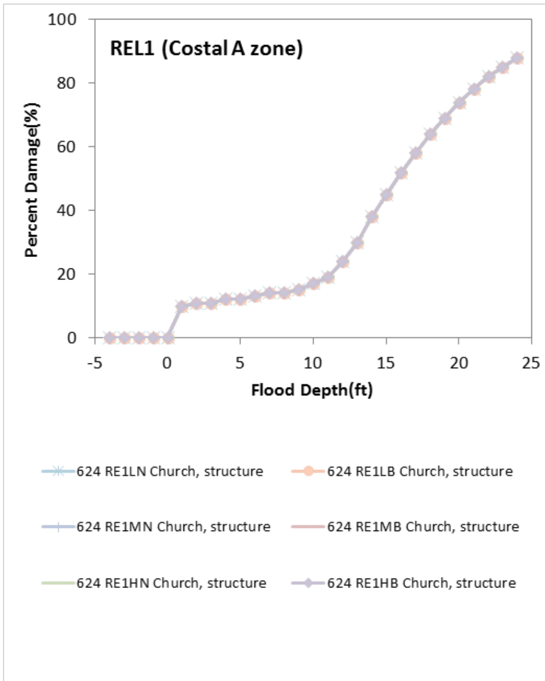
</td></tr>
<tr><td>


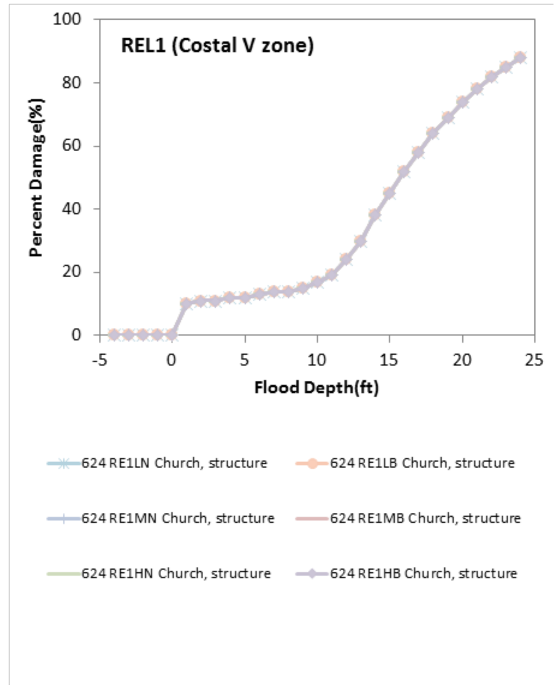
</td><td>


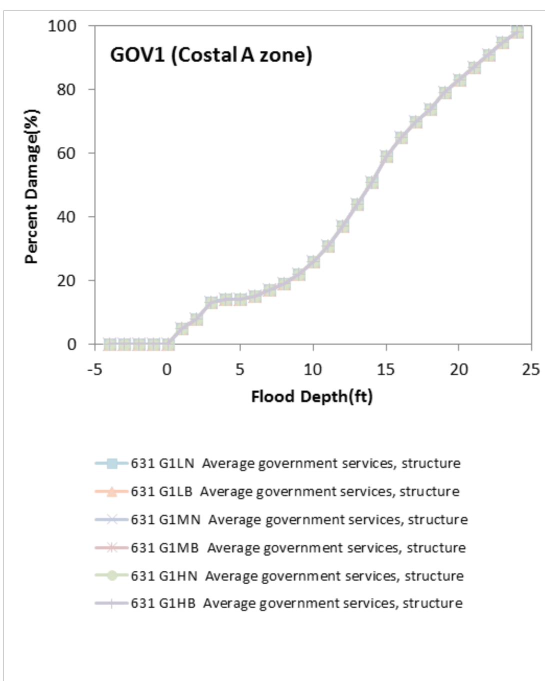
</td></tr>
</table>

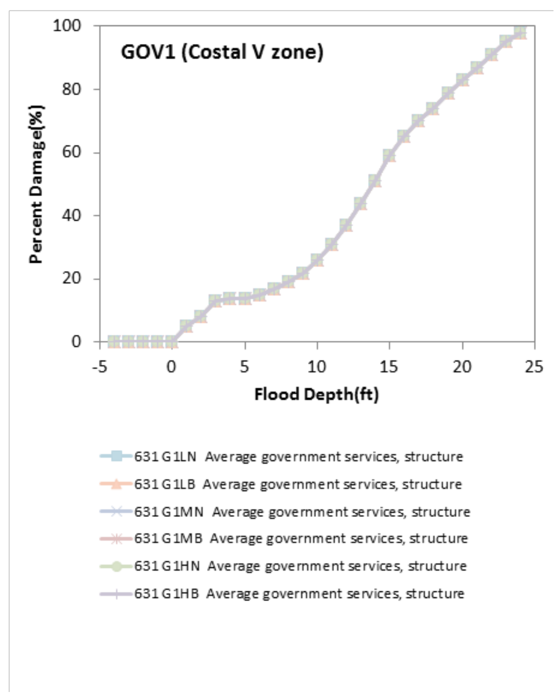
A zone curvs	V zone curves
 	
 	

A zone curvs	V zone curves

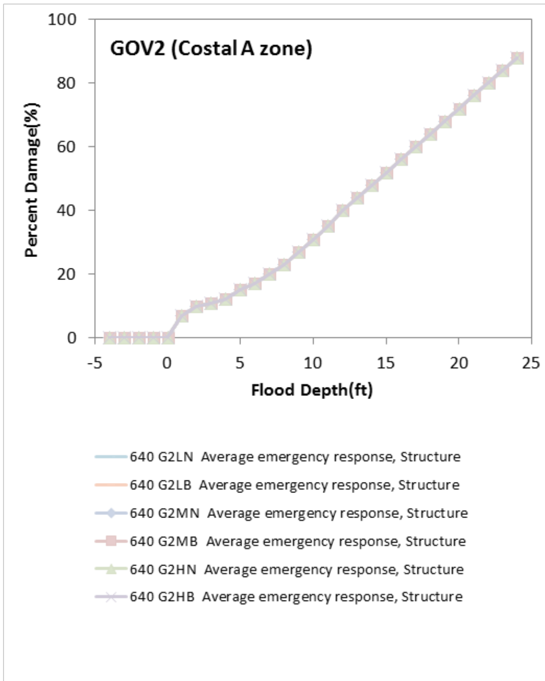


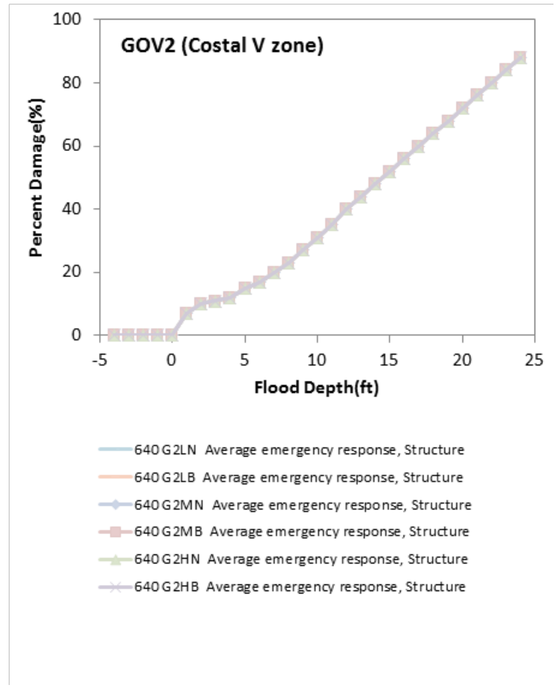


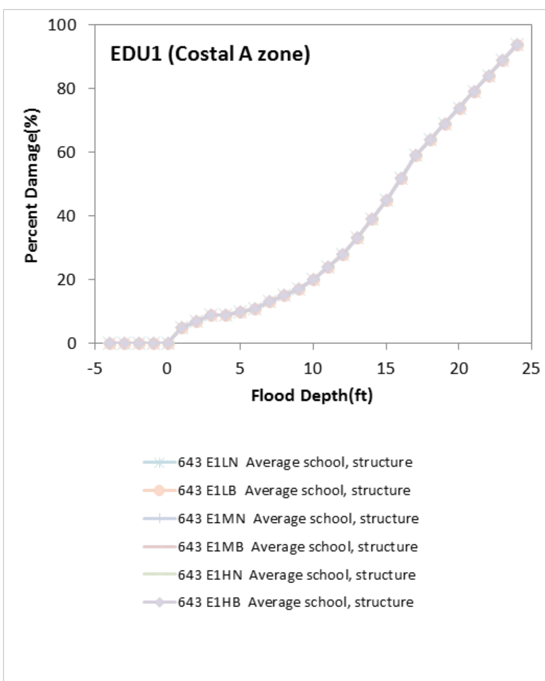


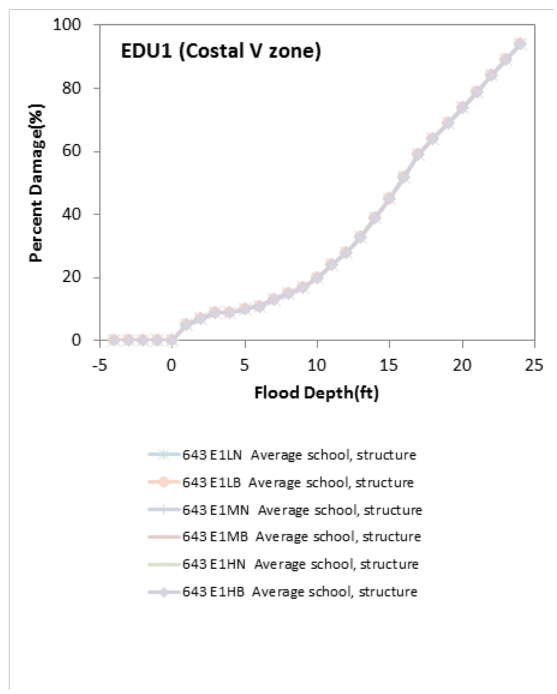


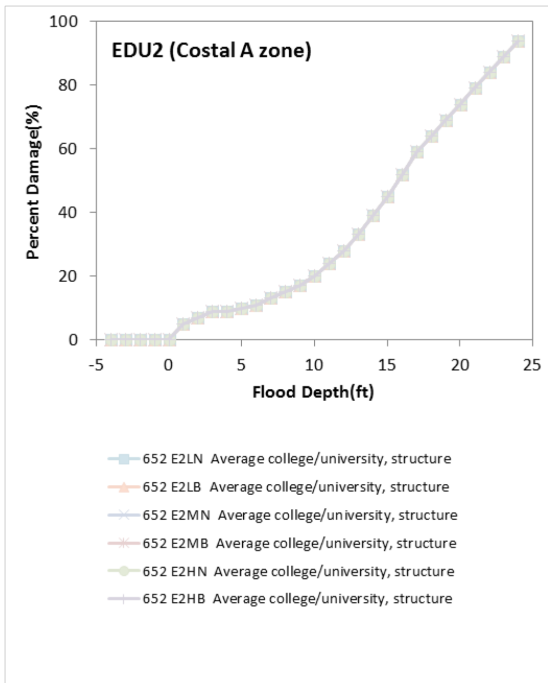
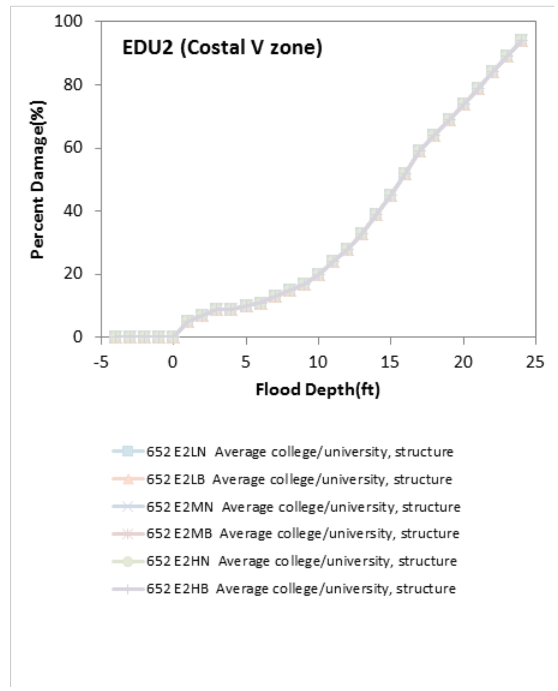
A zone curvs	V zone curves









A zone curvs	V zone curves
	

APPENDIX D 3-DIGIT NAICS CODE AND BUSINESS DESCRIPTION

NAICS Code (2-digit)	NAICS Code (3-digit)	Business Description
11	111	Crop Production
	112	Animal Production and Aquaculture
	113	Forestry and Logging
	114	Fishing, Hunting and Trapping
	115	Support Activities for Agriculture and Forestry
21	211	Oil and Gas Extraction
	212	Mining (except Oil and Gas)
	213	Support Activities for Mining
22	221	Utilities
23	236	Construction of Buildings
	237	Heavy and Civil Engineering Construction
	238	Specialty Trade Contractors
31	311	Food Manufacturing
	312	Beverage and Tobacco Product Manufacturing
	313	Textile Mills
	314	Textile Product Mills
	315	Apparel Manufacturing
	316	Leather and Allied Product Manufacturing
32	321	Wood Product Manufacturing
	322	Paper Manufacturing
	323	Printing and Related Support Activities
	324	Petroleum and Coal Products Manufacturing
	325	Chemical Manufacturing
	326	Plastics and Rubber Products Manufacturing
	327	Nonmetallic Mineral Product Manufacturing
33	331	Primary Metal Manufacturing
	332	Fabricated Metal Product Manufacturing
	333	Machinery Manufacturing
	334	Computer and Electronic Product Manufacturing

NAICS Code (2-digit)	NAICS Code (3-digit)	**Business Description**
	335	Electrical Equipment, Appliance, and Component Manufacturing
	336	Transportation Equipment Manufacturing
	337	Furniture and Related Product Manufacturing
	339	Miscellaneous Manufacturing
42	423	Merchant Wholesalers, Durable Goods
	424	Merchant Wholesalers, Nondurable Goods
	425	Wholesale Electronic Markets and Agents and Brokers
44	441	Motor Vehicle and Parts Dealers
	442	Furniture and Home Furnishings Stores
	443	Electronics and Appliance Stores
	444	Building Material and Garden Equipment and Supplies Dealers
	445	Food and Beverage Stores
	446	Health and Personal Care Stores
	447	Gasoline Stations
	448	Clothing and Clothing Accessories Stores
45	451	Sporting Goods, Hobby, Musical Instrument, and Book Stores
	452	General Merchandise Stores
	453	Miscellaneous Store Retailers
	454	Nonstore Retailers
48	481	Air Transportation
	482	Rail Transportation
	483	Water Transportation
	484	Truck Transportation
	485	Transit and Ground Passenger Transportation
	486	Pipeline Transportation
	487	Scenic and Sightseeing Transportation
	488	Support Activities for Transportation
49	491	Postal Service
	492	Couriers and Messengers
	493	Warehousing and Storage
51	511	Publishing Industries (except Internet)

NAICS Code (2-digit)	NAICS Code (3-digit)	**Business Description**
	512	Motion Picture and Sound Recording Industries
	515	Broadcasting (except Internet)
	517	Telecommunications
	518	Data Processing, Hosting, and Related Services
	519	Other Information Services
52	521	Monetary Authorities-Central Bank
	522	Credit Intermediation and Related Activities
	523	Securities, Commodity Contracts, and Other Financial Investments and Related Activities
	524	Insurance Carriers and Related Activities
	525	Funds, Trusts, and Other Financial Vehicle
53	531	Real Estate
	532	Rental and Leasing Services
	533	Lessors of Nonfinancial Intangible Assets (except Copyrighted Works)
54	541	Professional, Scientific, and Technical Services
55	551	Management of Companies and Enterprises
56	561	Administrative and Support Services
	562	Waste Management and Remediation Services
61	611	Educational Services
62	621	Ambulatory Health Care Services
	622	Hospitals
	623	Nursing and Residential Care Facilities
	624	Social Assistance
71	711	Performing Arts, Spectator Sports, and Related Industries
	712	Museums, Historical Sites, and Similar Institutions
	713	Amusement, Gambling, and Recreation Industries
72	721	Accommodation
	722	Food Services and Drinking Places
81	811	Repair and Maintenance
	812	Personal and Laundry Services
	813	Religious, Grantmaking, Civic, Professional, and Similar Organizations

NAICS Code (2-digit)	NAICS Code (3-digit)	Business Description
	814	Private Households
92	921	Executive, Legislative, and Other General Government Support
	922	Justice, Public Order, and Safety Activities
	923	Administration of Human Resource Programs
	924	Administration of Environmental Quality Programs
	925	Administration of Housing Programs, Urban Planning, and Community Development
	926	Administration of Economic Programs
	927	Space Research and Technology
	928	National Security and International Affairs

APPENDIX E SRF ZONE GENERATION USING ARCGIS

Geoprocessing software ArcGIS (Version 10.1) is used to conduct the SRF zone delineation. Ideally, the SRF stations are converted into raster format and used as the "pour points"[3], and then by executing the "watershed delineation" function in ArcGIS toolbox, the SRF zones will be created automatically. However, if we use the SRF stations directly as the pour points, ArcGIS will generate small polygons which do not cover the entire area. To deal with this, instead of using the SRF stations, the segments of coastline closest to each SRF station are used as the outlets of the "watersheds". Following are the steps of generating the SRF zones using this alternative approach:

(1) Reclassify the DEM as the "land" and "ocean". (See Figure E-1)

(2) Convert the classified DEM into polygons.

(3) Use the SRF stations to create Thiessen polygons. (See Figure E-2)

(4) Use the Theissen polygons to subdivide the "ocean" polygon. (See Figure E-3)

(5) Convert the subdivided "ocean" polygon into raster format.

(6) Use the raster data created in (5) as the "pour points" to delineate the SRF zones. (See Figure E-4)

The resulting SRF zones for Corpus Christi and Gulfport are shown in Figure E-4 through Figure E-7.

[3] ArcGIS terminology, a parameter needed for watershed delineation

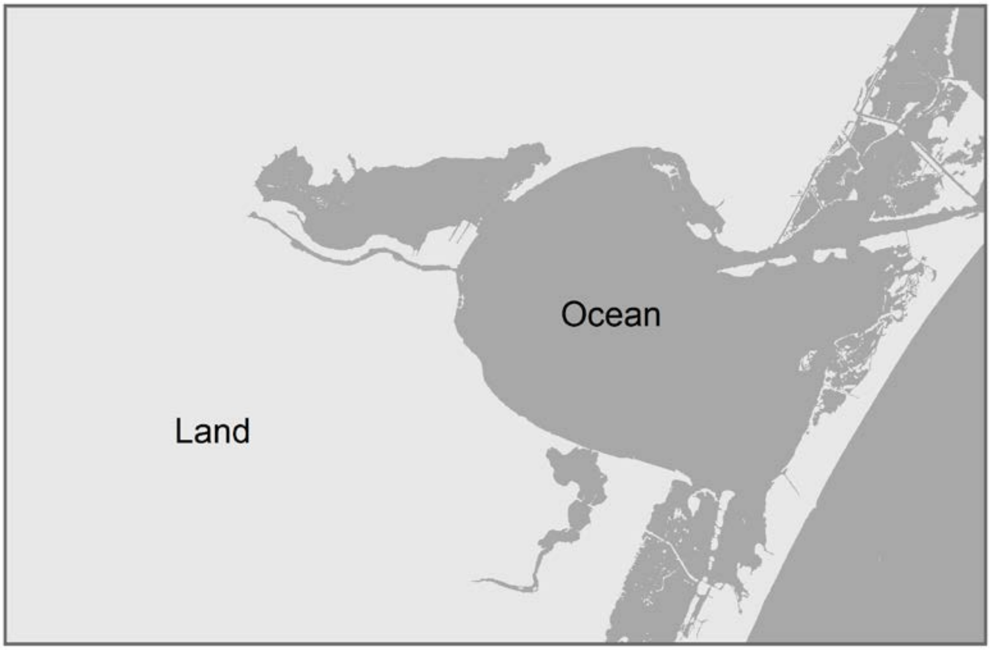


Figure E-1 Reclassify the DEM into Two Parts (Land and Ocean)

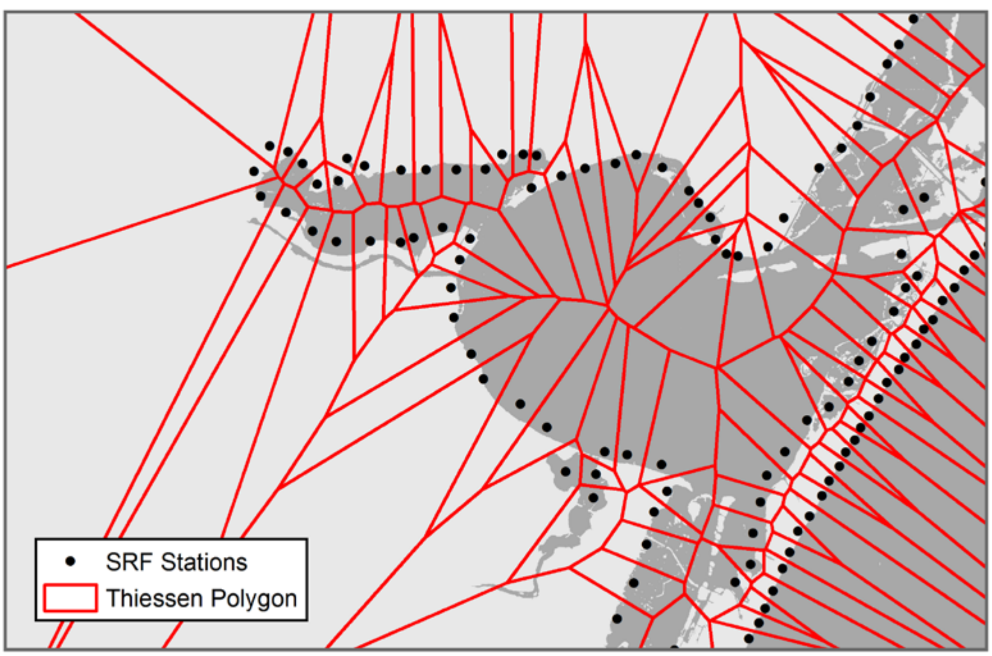


Figure E-2 Theissen Polygon Based on SRF Stations

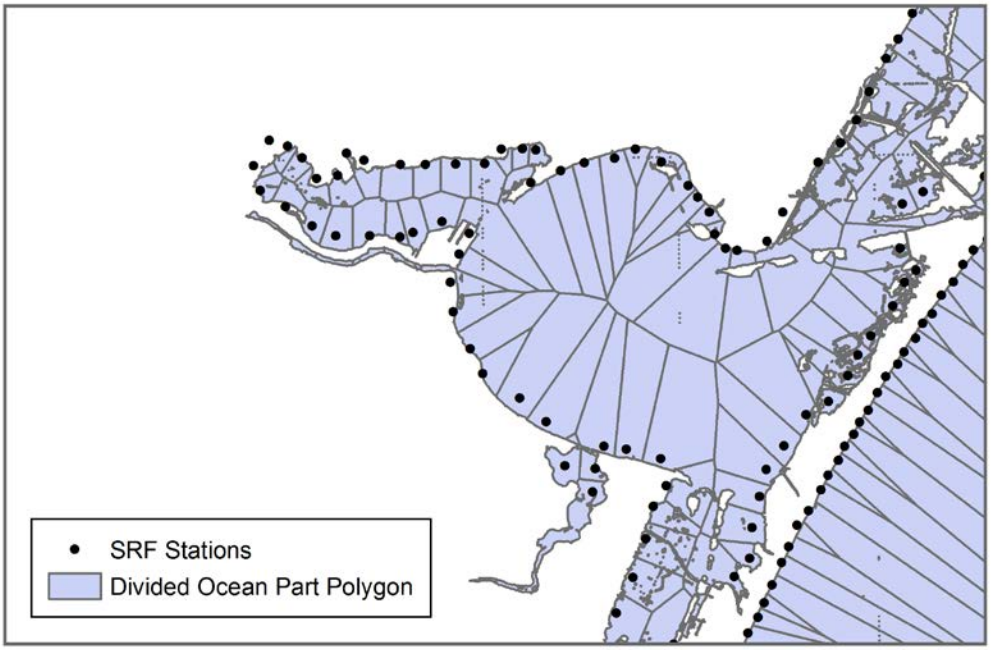


Figure E-3 Divided Ocean Part Polygons

Figure E-4 Finalized SRF Zones (Corpus Christi, TX)

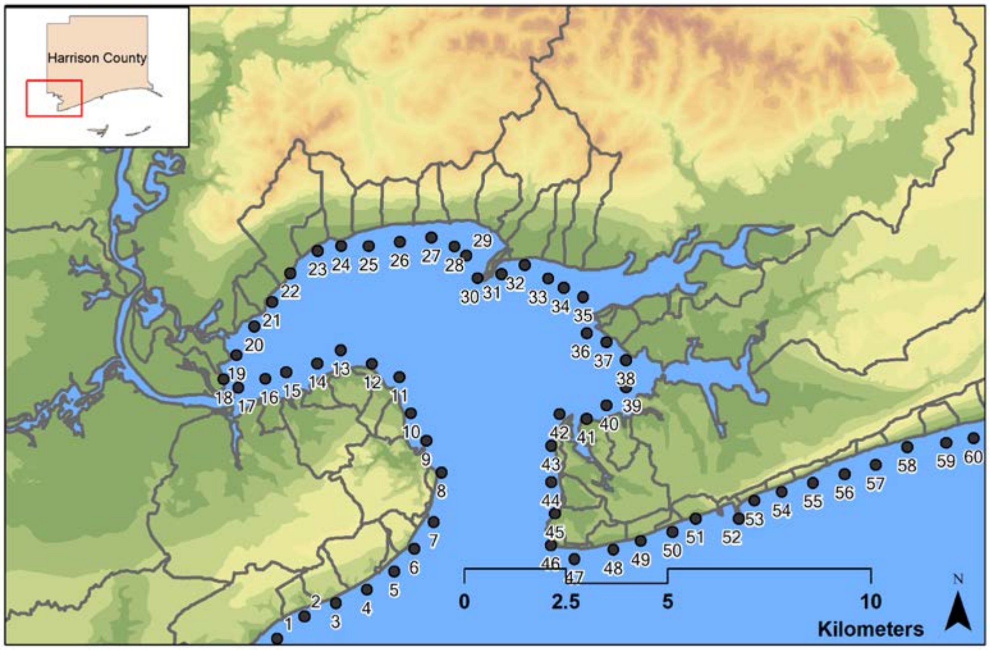


Figure E-5 Finalized SRF Zones (Gulfport, MS; Bay St. Louis)

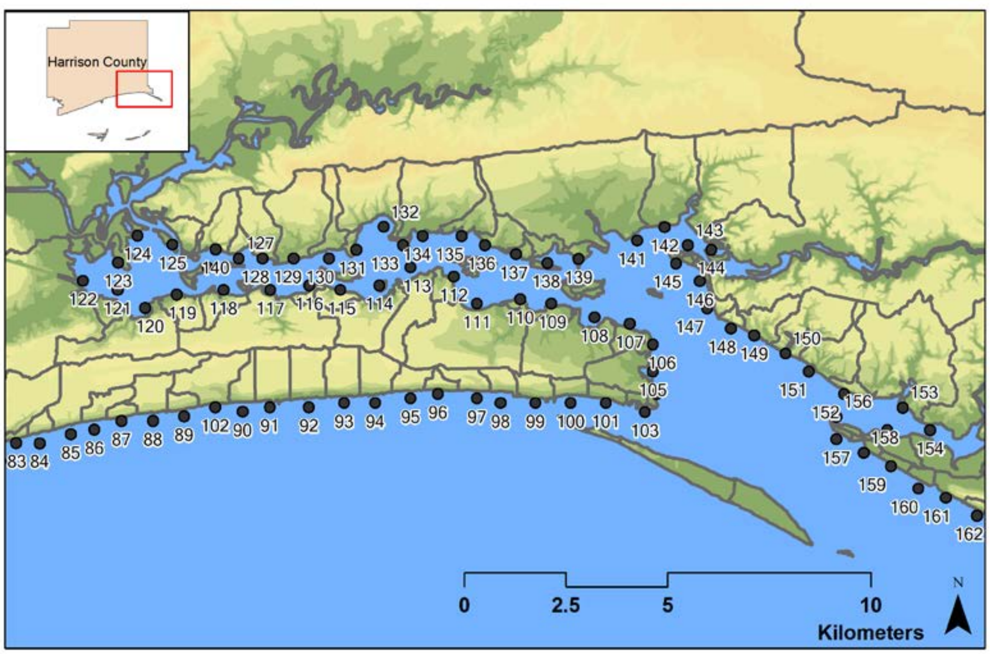


Figure E-6 Finalized SRF Zones (Gulfport, MS; Biloxi Bay)

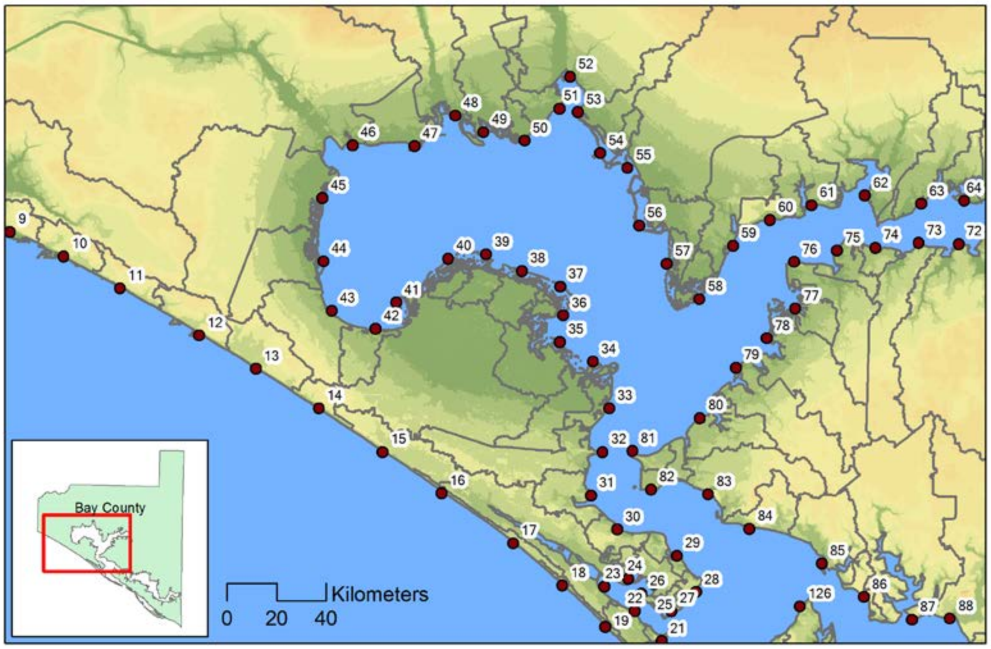
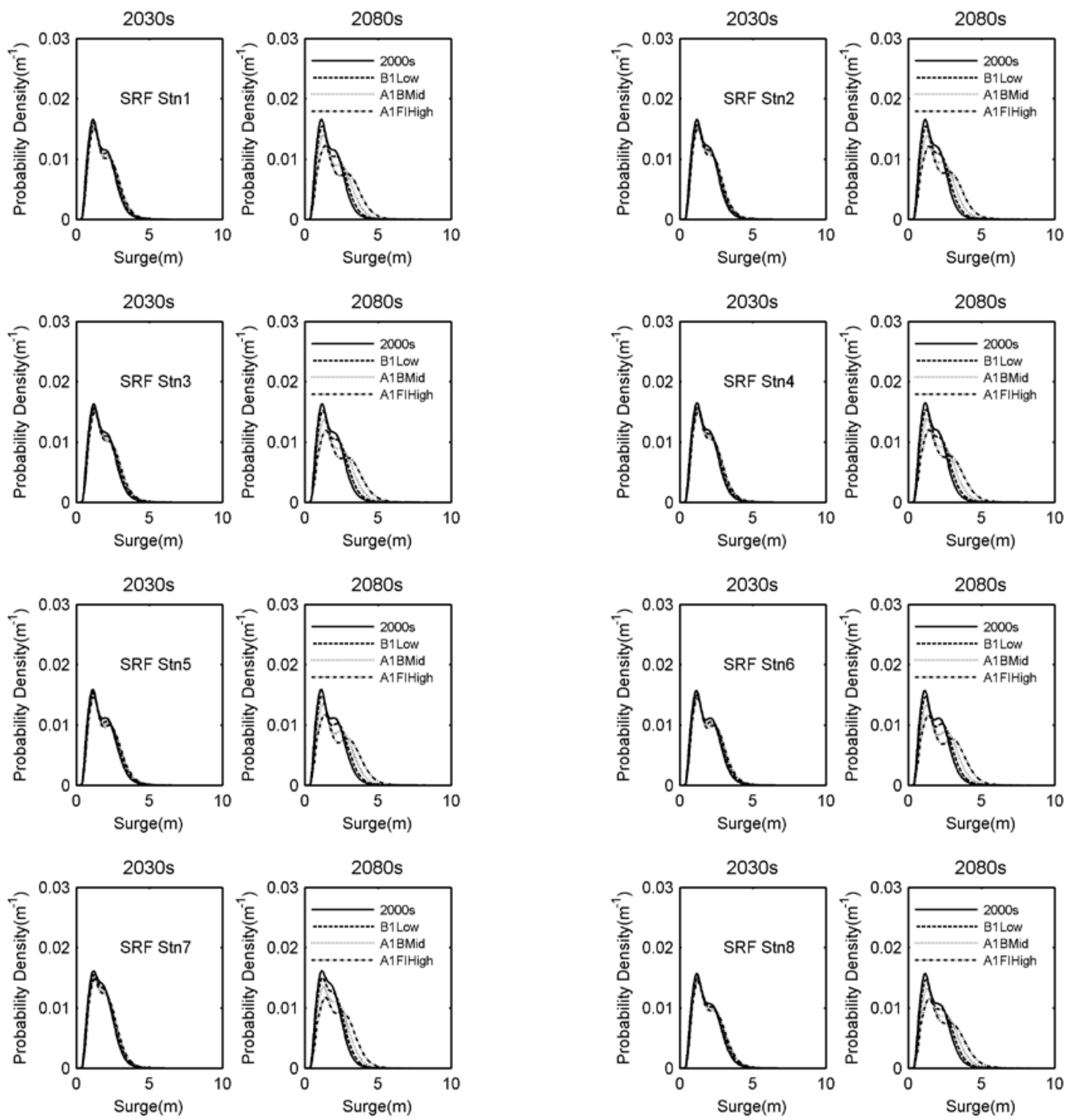
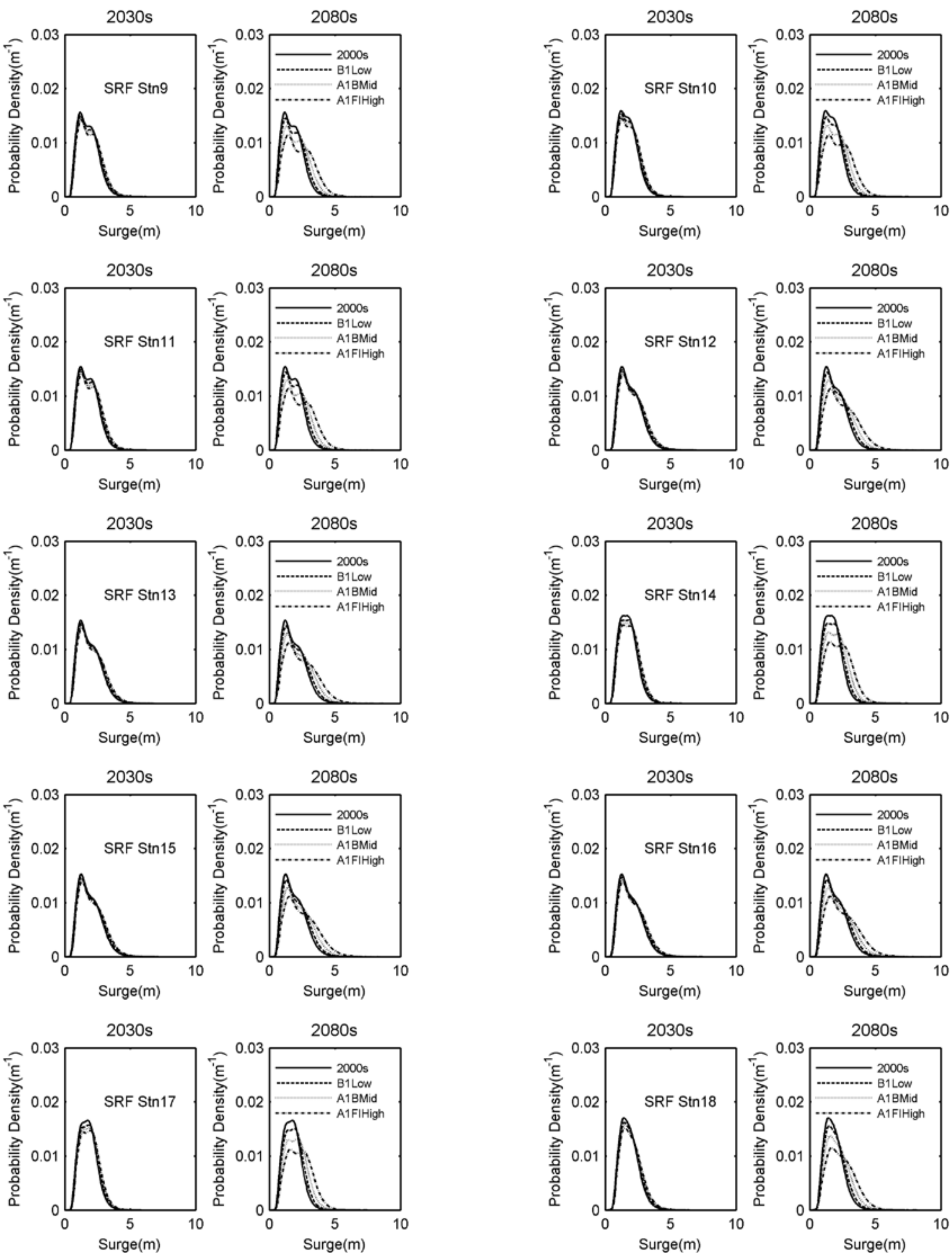


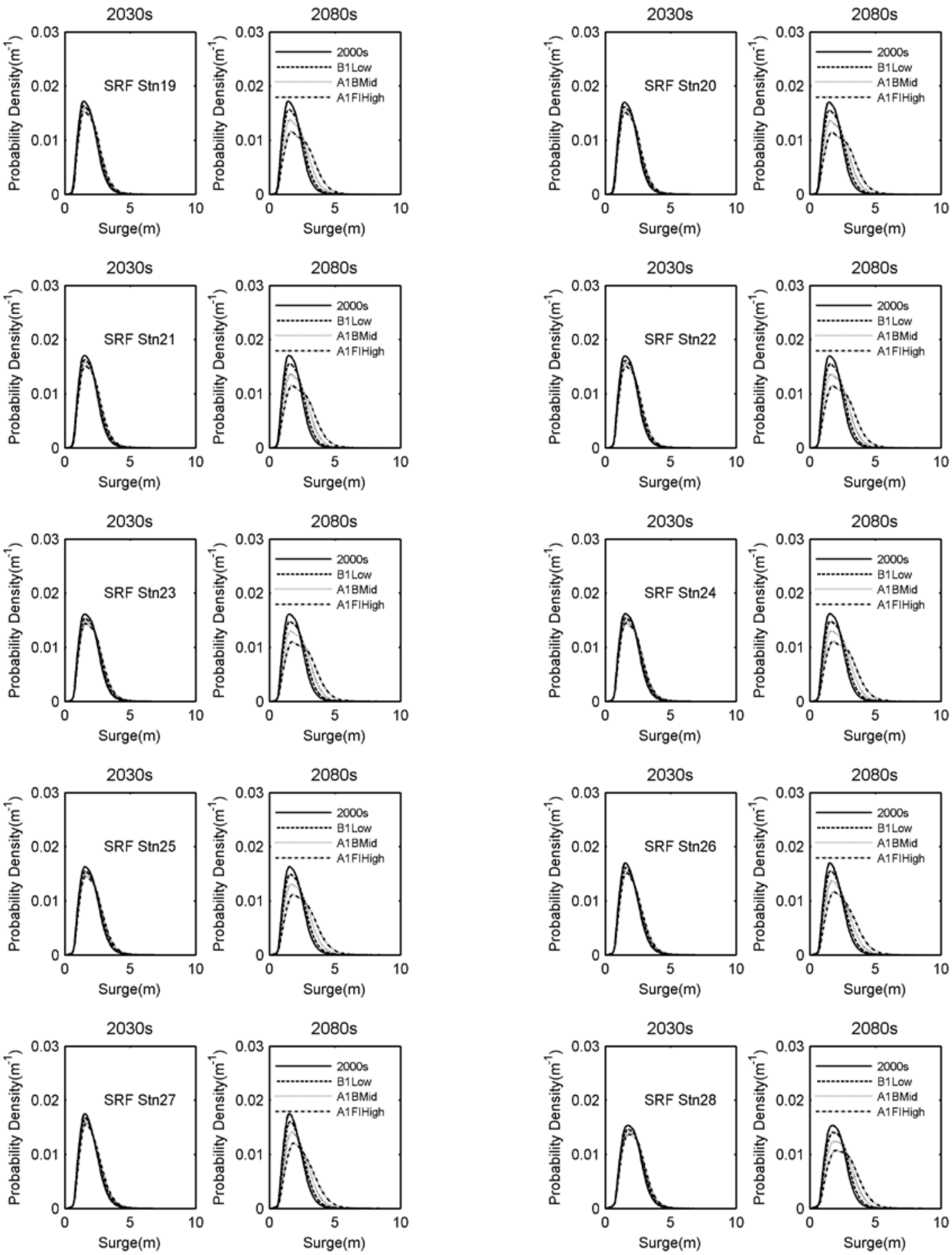
Figure E-7 Finalized SRF Zones (Panama, FL)

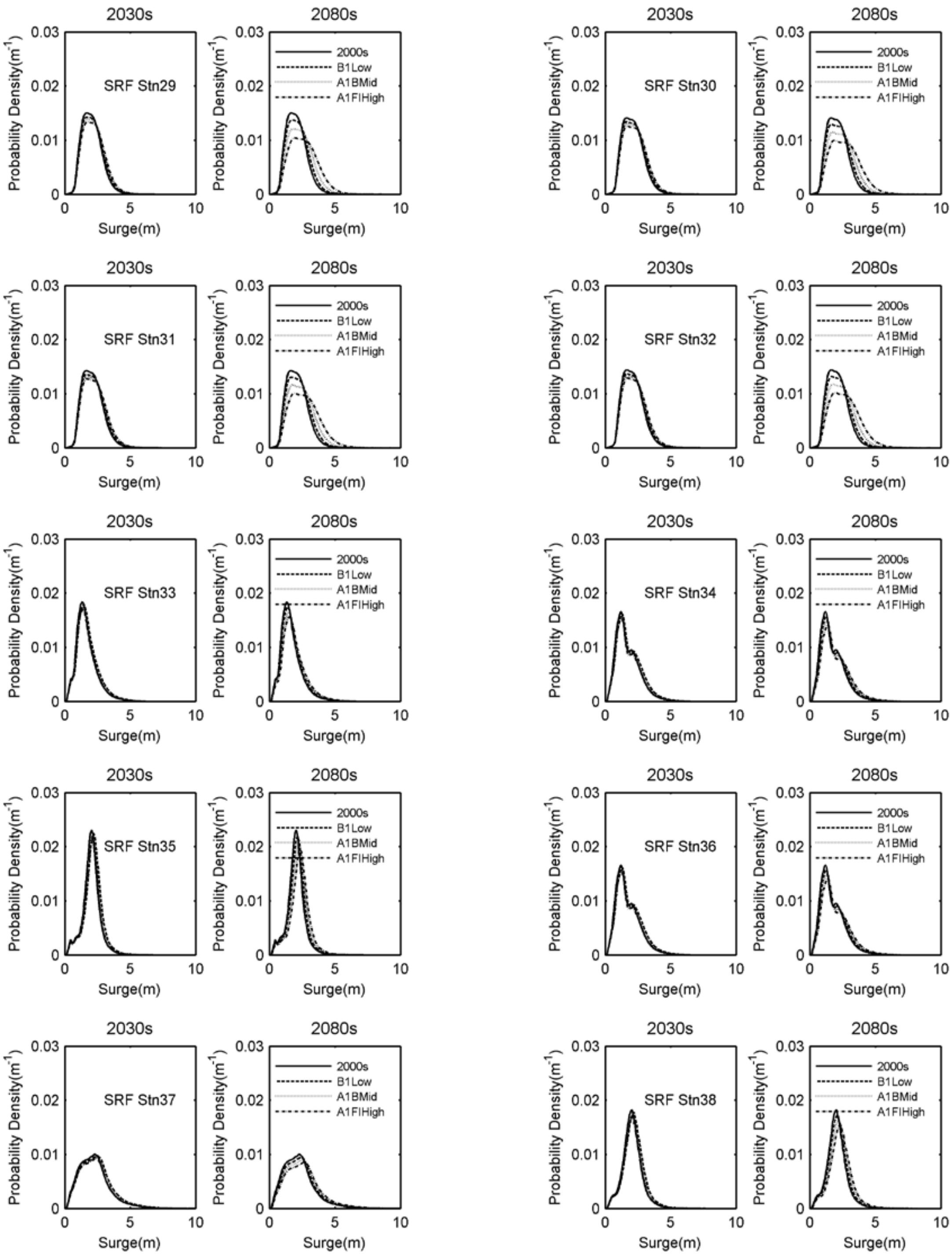
APPENDIX F SURGE PROBABILITY DENSITY FUNCTIONS FOR ALL SRFS

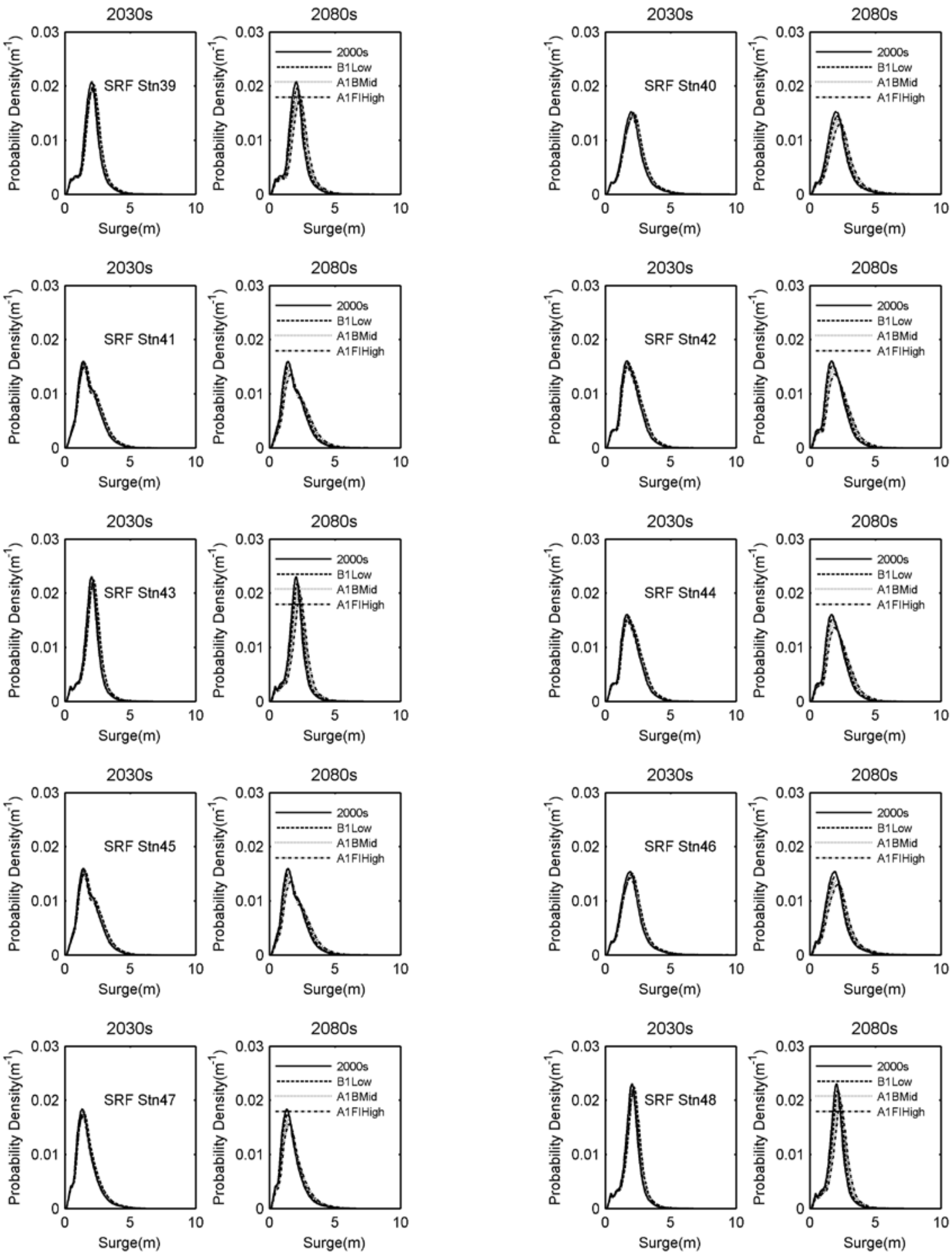
F.1. Corpus Christi, TX

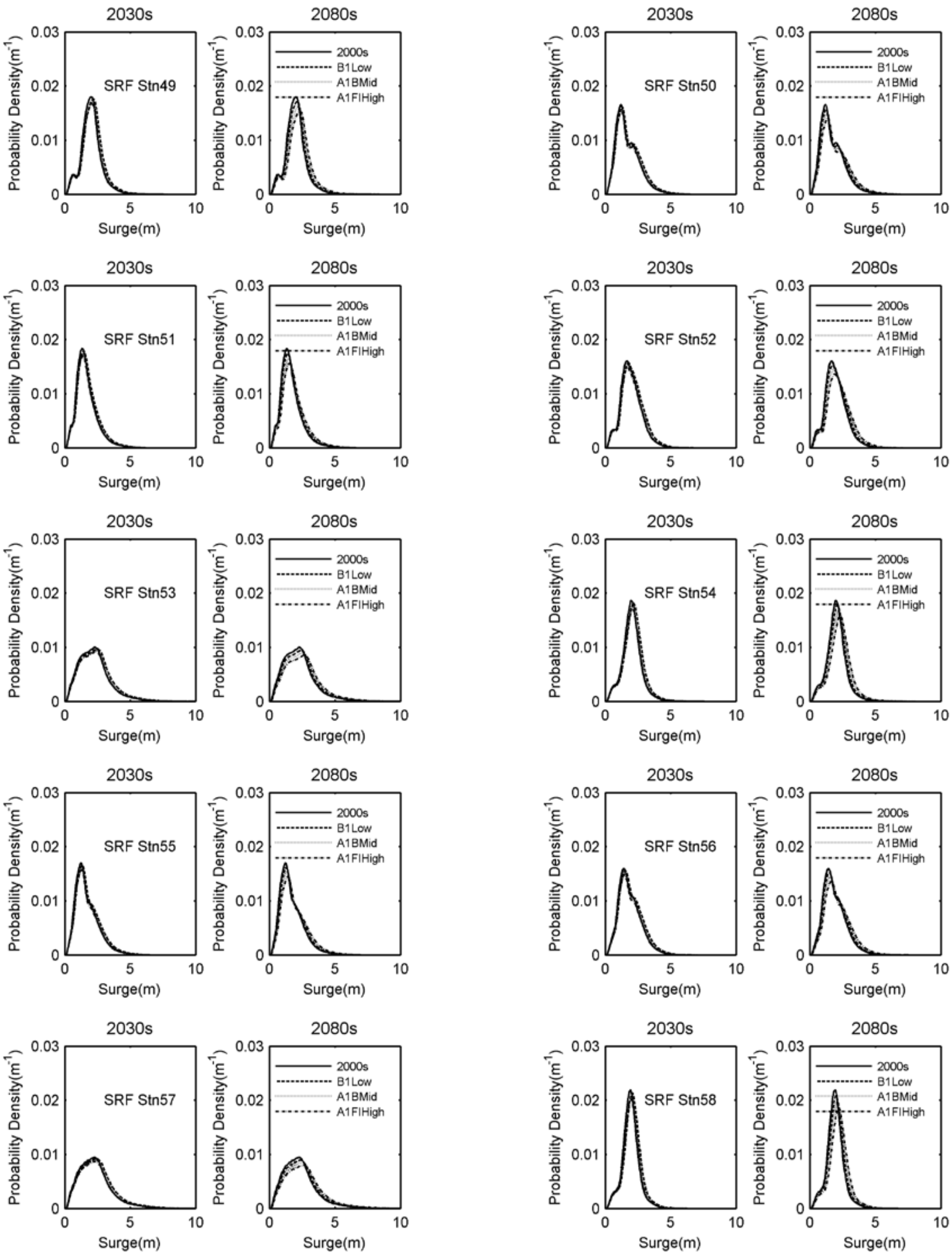


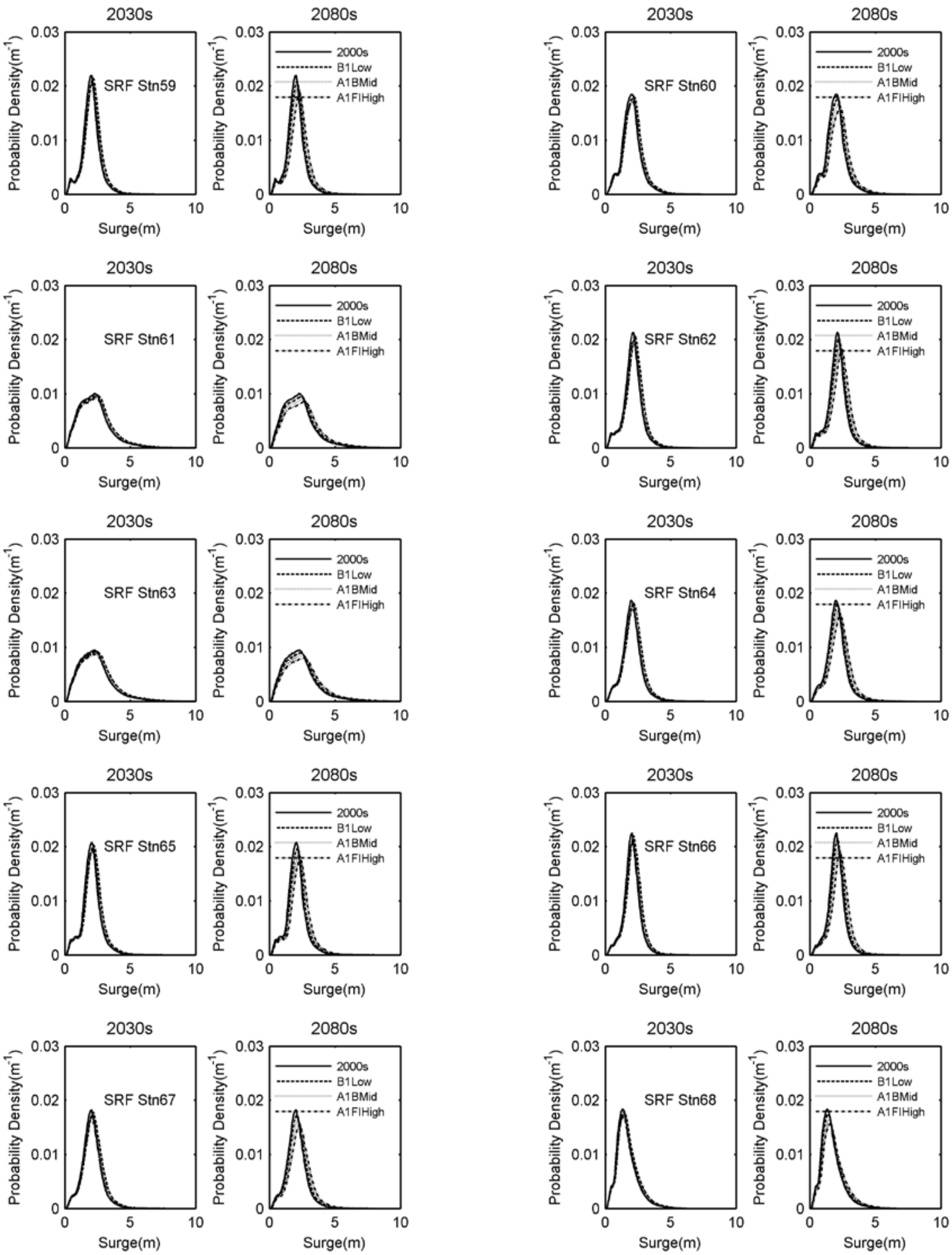

2030s
2080s
SRF Stn9
SRF Stn10
SRF Stn11
SRF Stn12
SRF Stn13
SRF Stn14
SRF Stn15
SRF Stn16
SRF Stn17
SRF Stn18
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)
0
0.01
0.02
0.03
5
10

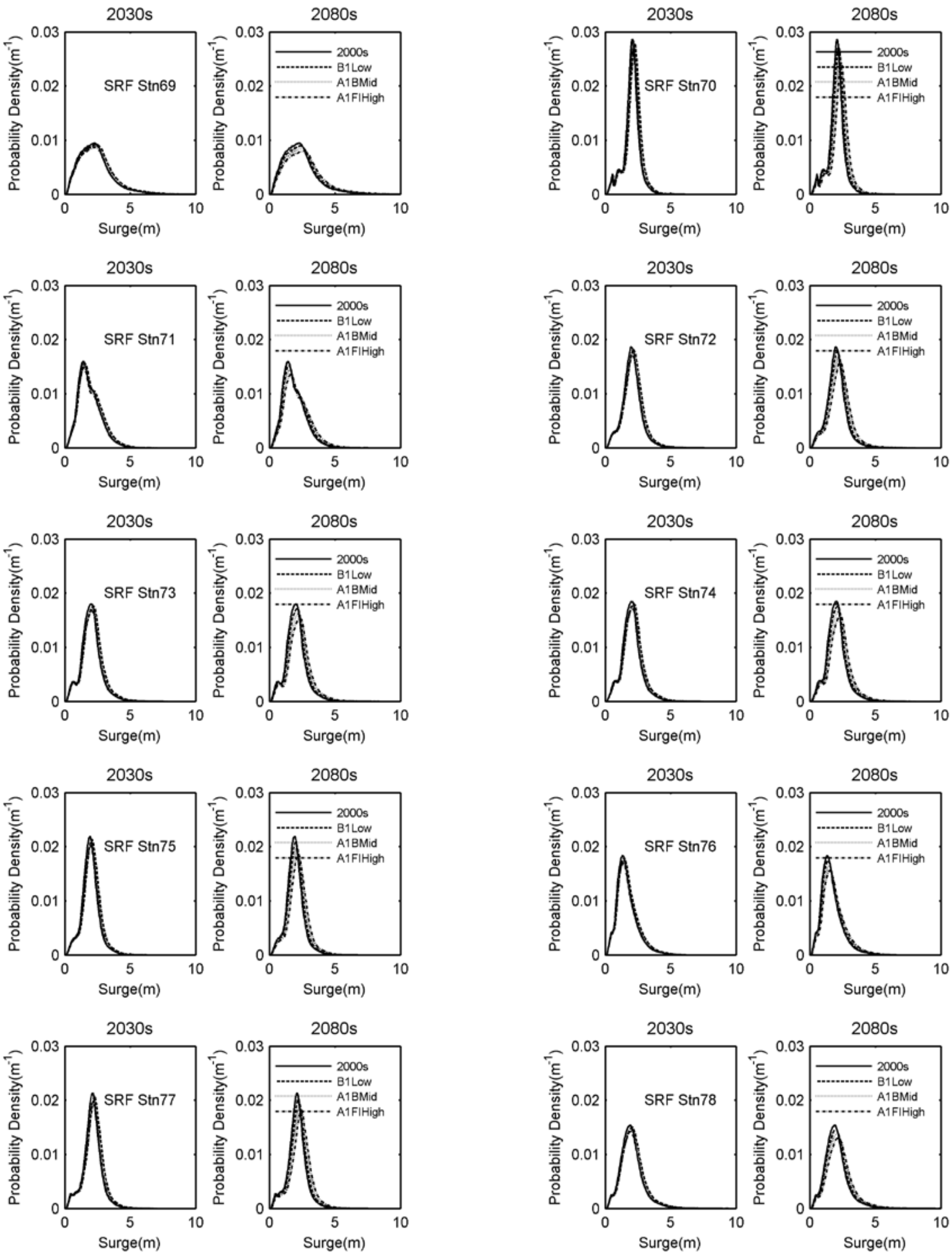

2030s
2080s
SRF Stn19
SRF Stn20
SRF Stn21
SRF Stn22
SRF Stn23
SRF Stn24
SRF Stn25
SRF Stn26
SRF Stn27
SRF Stn28
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m-1)
Surge(m)
0
0.01
0.02
0.03
5
10

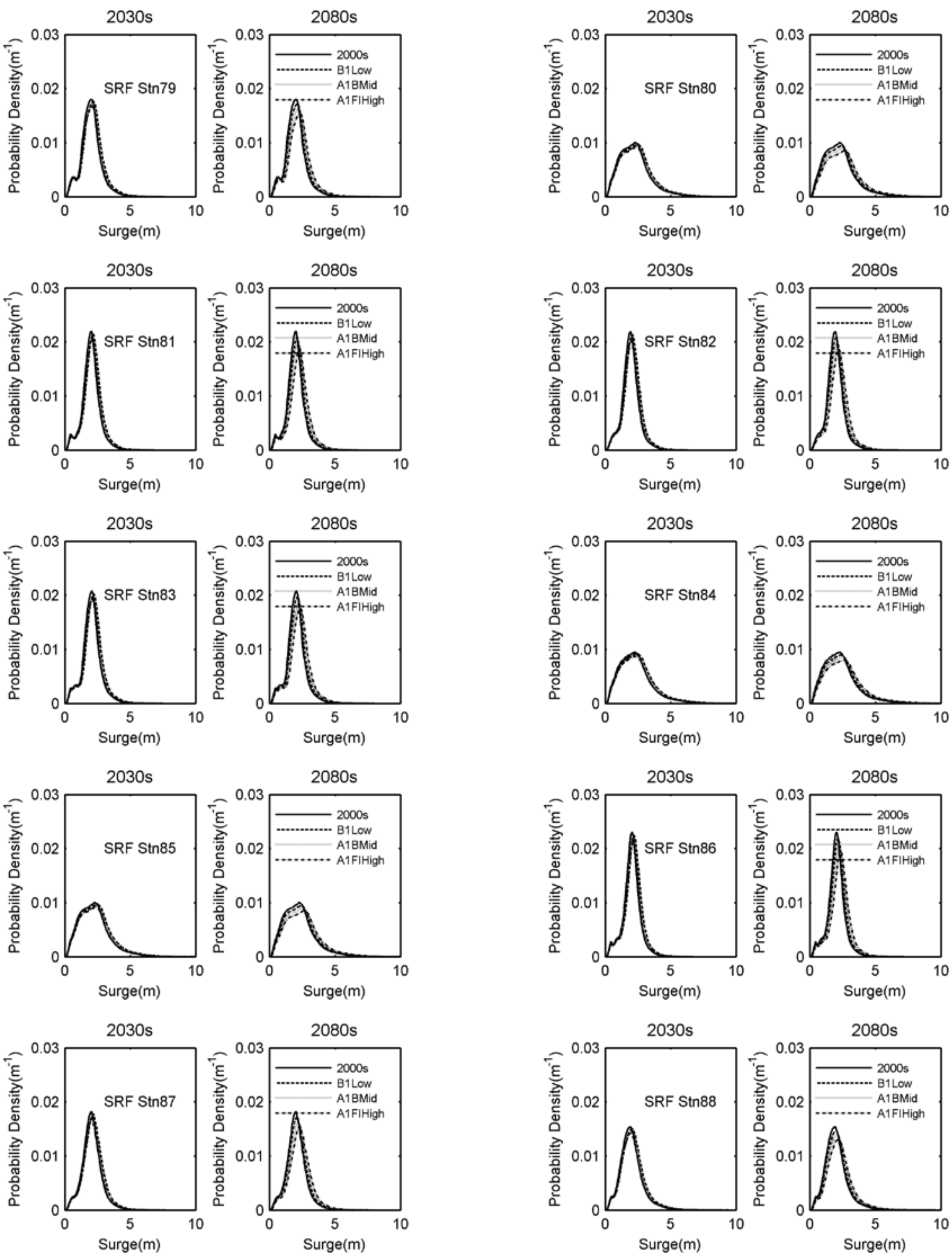

2030s
2080s
SRF Stn29
SRF Stn30
SRF Stn31
SRF Stn32
SRF Stn33
SRF Stn34
SRF Stn35
SRF Stn36
SRF Stn37
SRF Stn38
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m-1)
Surge(m)


2030s
2080s
SRF Stn39
SRF Stn40
SRF Stn41
SRF Stn42
SRF Stn43
SRF Stn44
SRF Stn45
SRF Stn46
SRF Stn47
SRF Stn48
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)

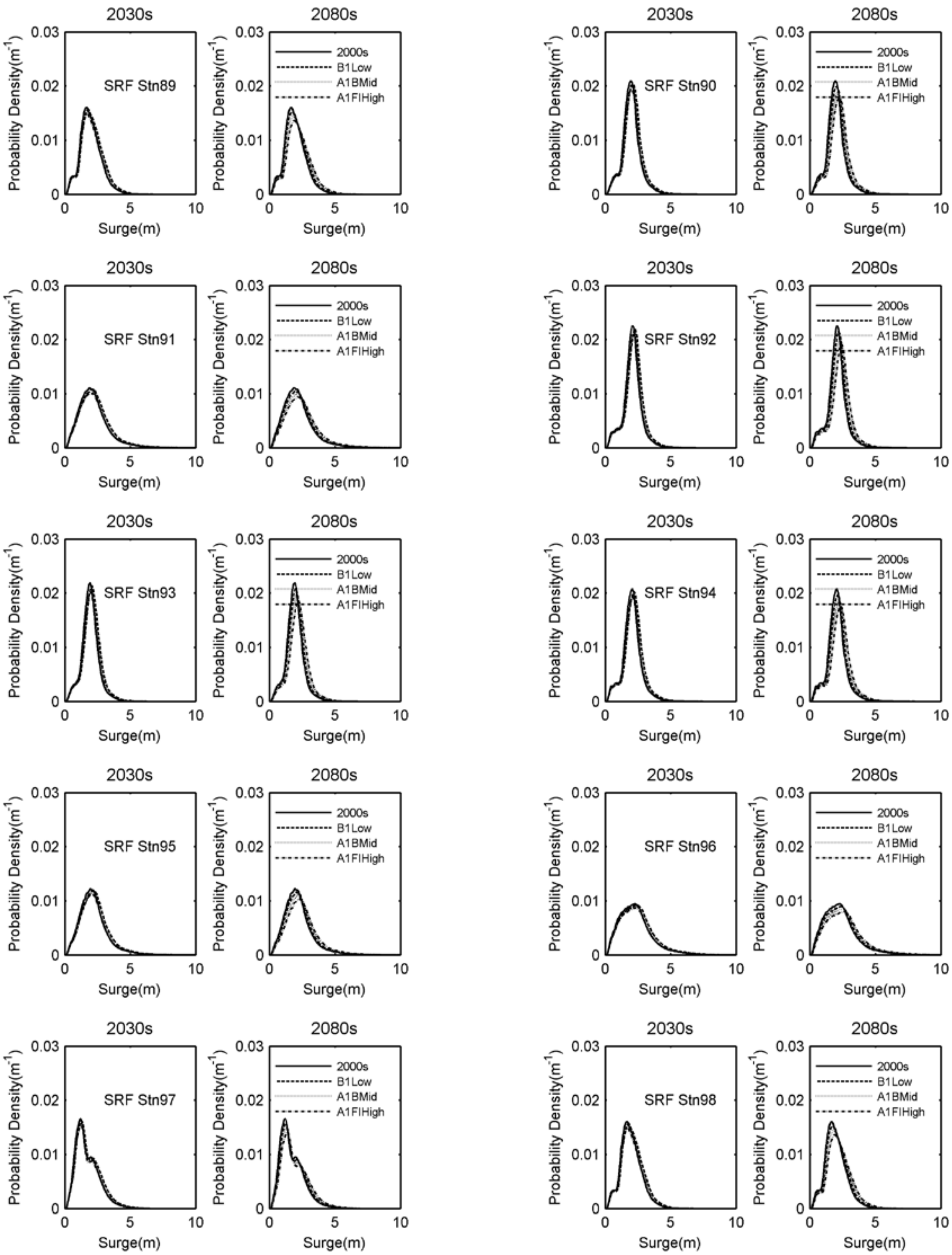

2030s
2080s
SRF Stn49
SRF Stn50
SRF Stn51
SRF Stn52
SRF Stn53
SRF Stn54
SRF Stn55
SRF Stn56
SRF Stn57
SRF Stn58
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)
0
0.01
0.02
0.03
0
5
10

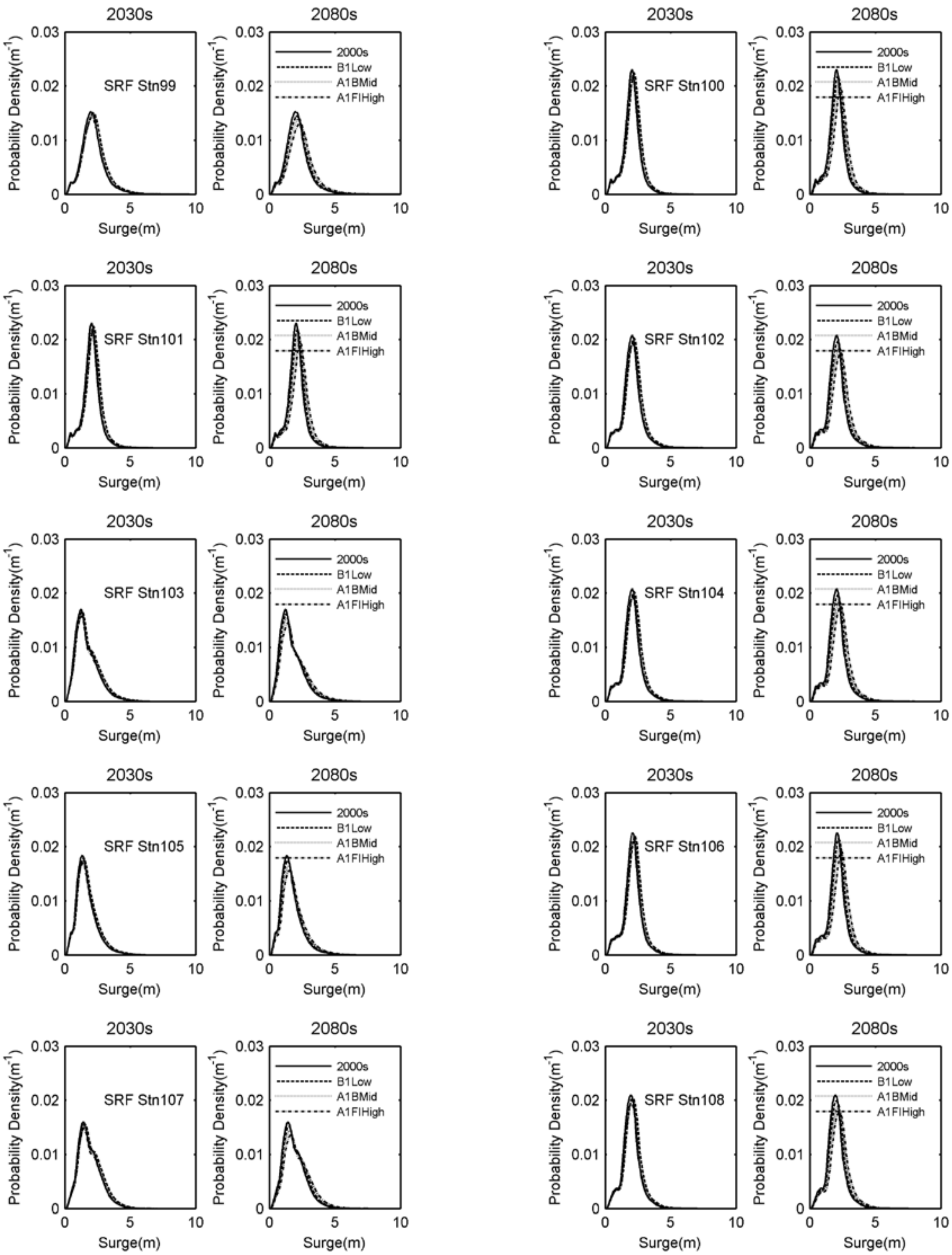

2030s
2080s
SRF Stn59
SRF Stn60
SRF Stn61
SRF Stn62
SRF Stn63
SRF Stn64
SRF Stn65
SRF Stn66
SRF Stn67
SRF Stn68
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)
0
0.01
0.02
0.03
5
10


2030s
2080s
Probability Density(m-1)
Surge(m)
SRF Stn69
SRF Stn70
SRF Stn71
SRF Stn72
SRF Stn73
SRF Stn74
SRF Stn75
SRF Stn76
SRF Stn77
SRF Stn78
2000s
B1Low
A1BMid
A1FIHigh

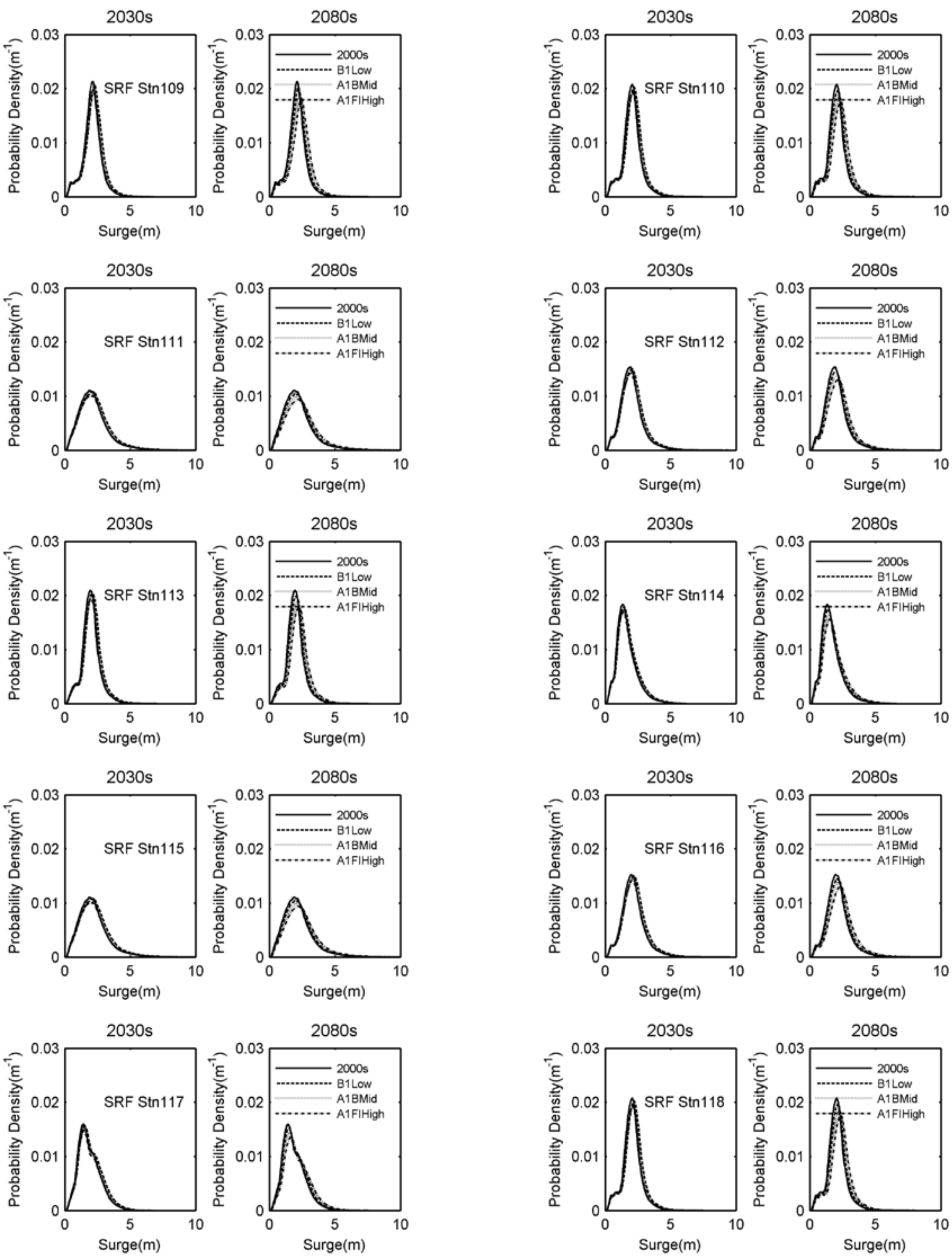


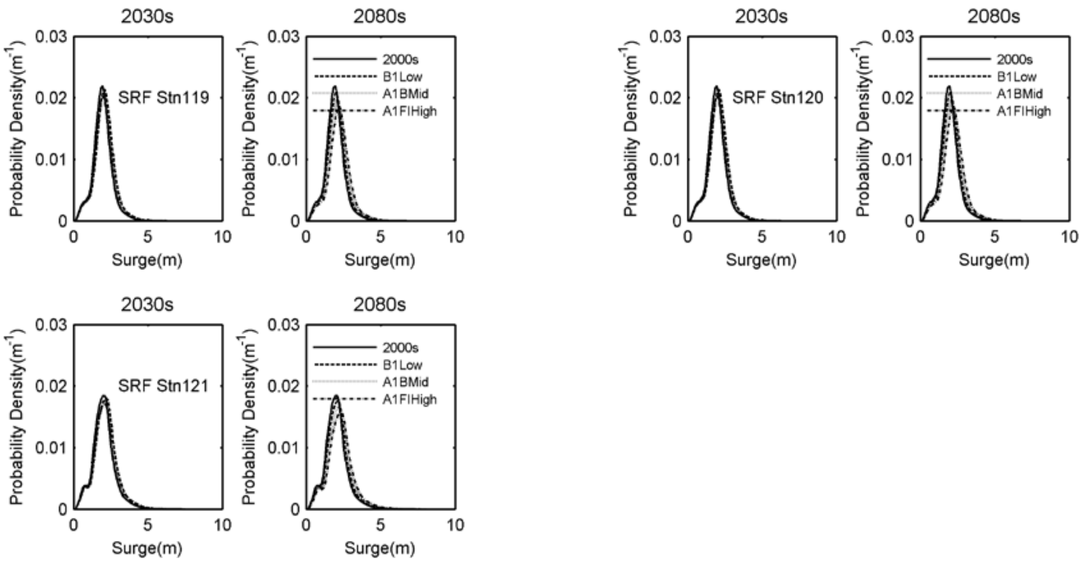
2030s
2080s
Probability Density(m-1)
Surge(m)
SRF Stn79
SRF Stn80
SRF Stn81
SRF Stn82
SRF Stn83
SRF Stn84
SRF Stn85
SRF Stn86
SRF Stn87
SRF Stn88
2000s
B1Low
A1BMid
A1FIHigh


2030s
2080s
Probability Density(m^{-1})
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh
SRF Stn89
SRF Stn90
SRF Stn91
SRF Stn92
SRF Stn93
SRF Stn94
SRF Stn95
SRF Stn96
SRF Stn97
SRF Stn98

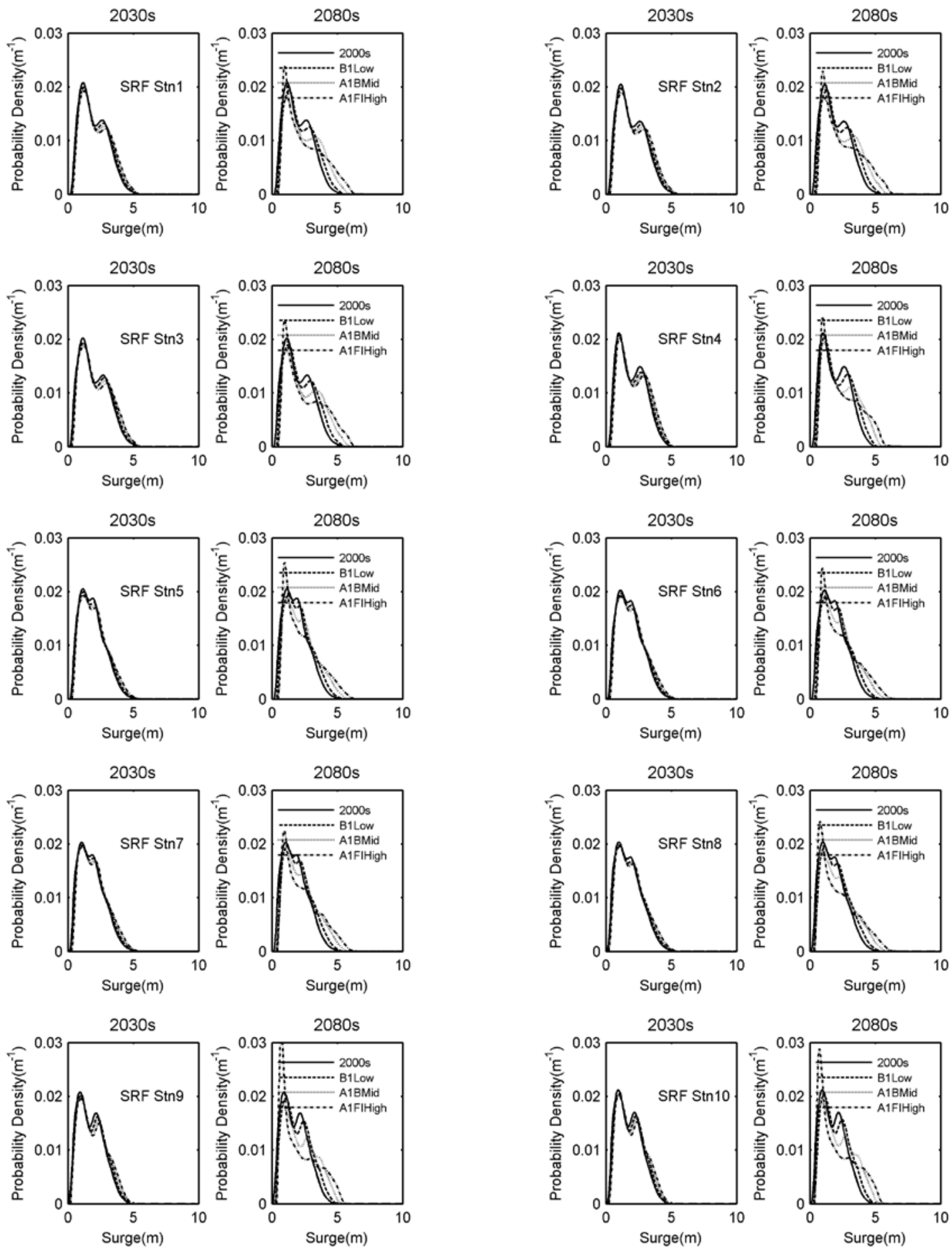


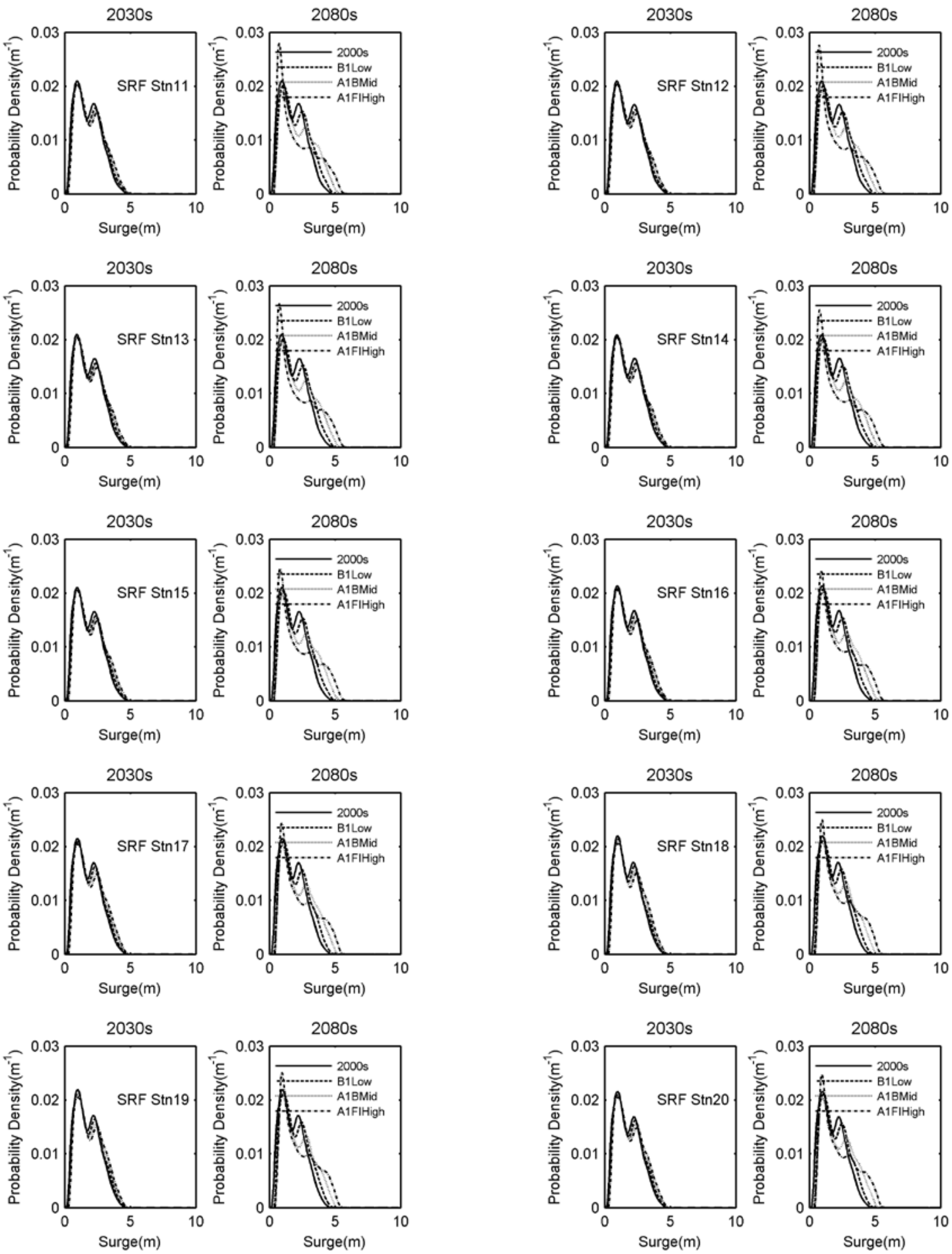
2030s
2080s
Probability Density(m^-1)
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh
SRF Stn99
SRF Stn100
SRF Stn101
SRF Stn102
SRF Stn103
SRF Stn104
SRF Stn105
SRF Stn106
SRF Stn107
SRF Stn108


2030s
2080s
Probability Density(m^{-1})
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh
SRF Stn109
SRF Stn110
SRF Stn111
SRF Stn112
SRF Stn113
SRF Stn114
SRF Stn115
SRF Stn116
SRF Stn117
SRF Stn118

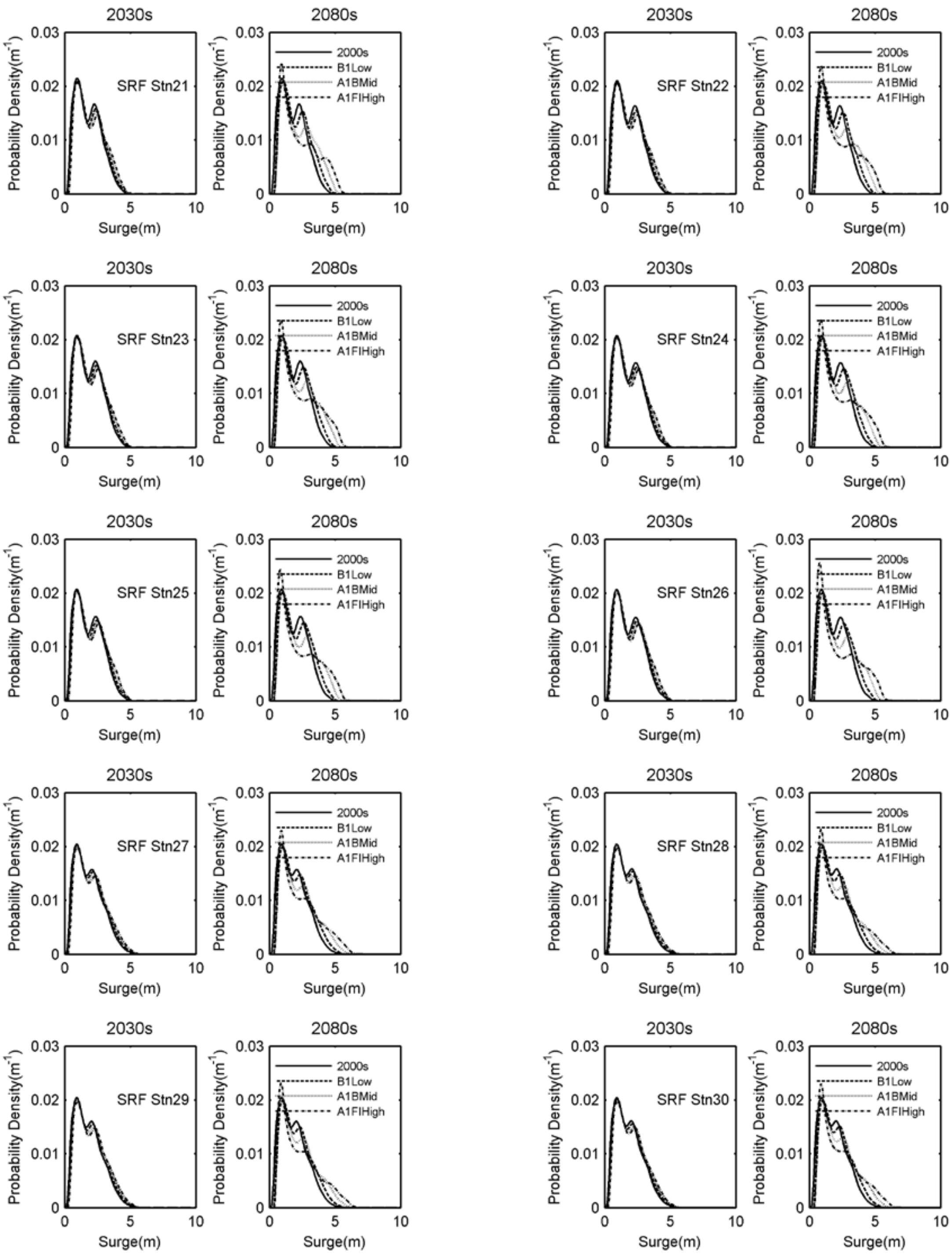

2030s
2080s
SRF Stn119
SRF Stn120
SRF Stn121
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m-1)
Surge(m)

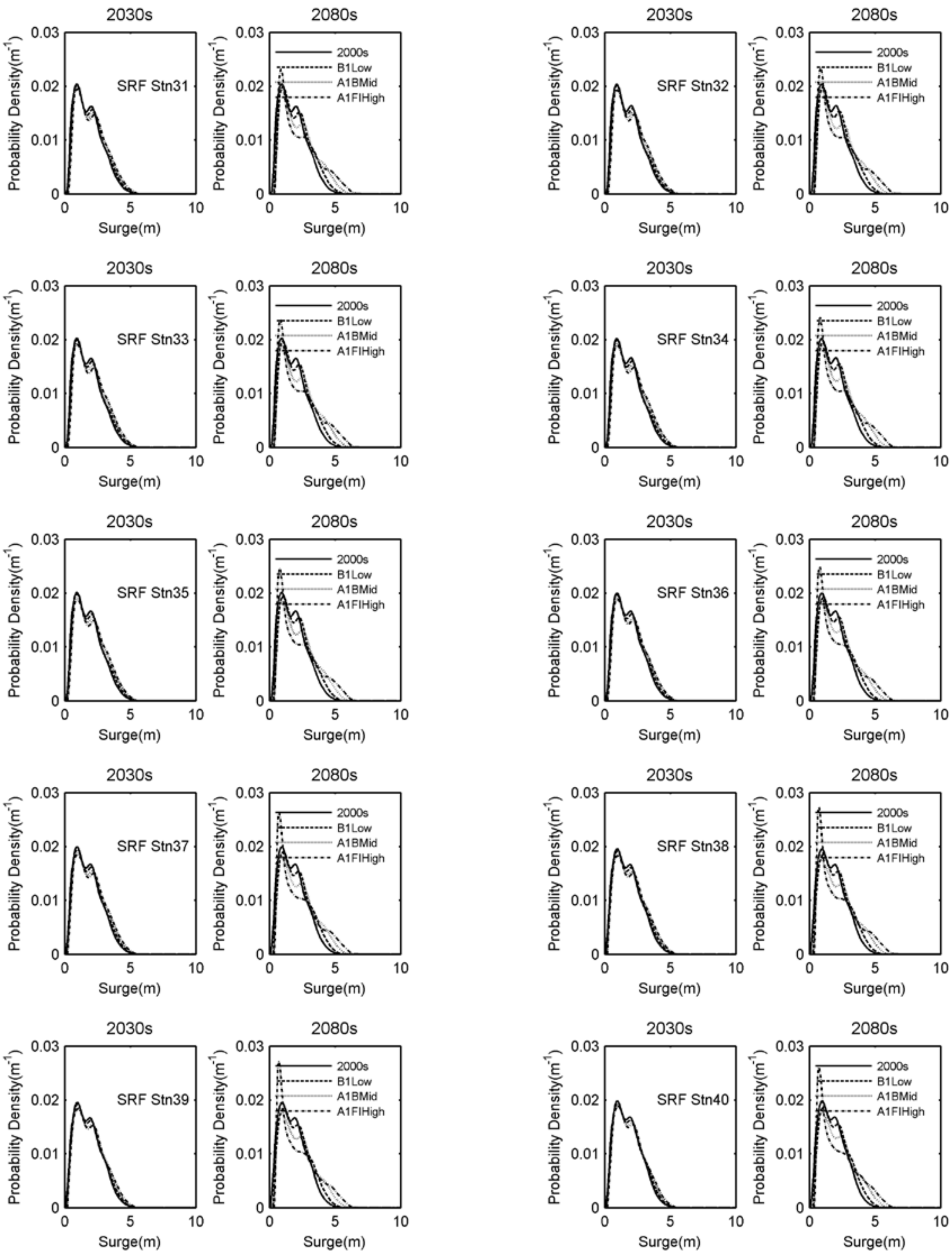
F.2. Gulfport, MS

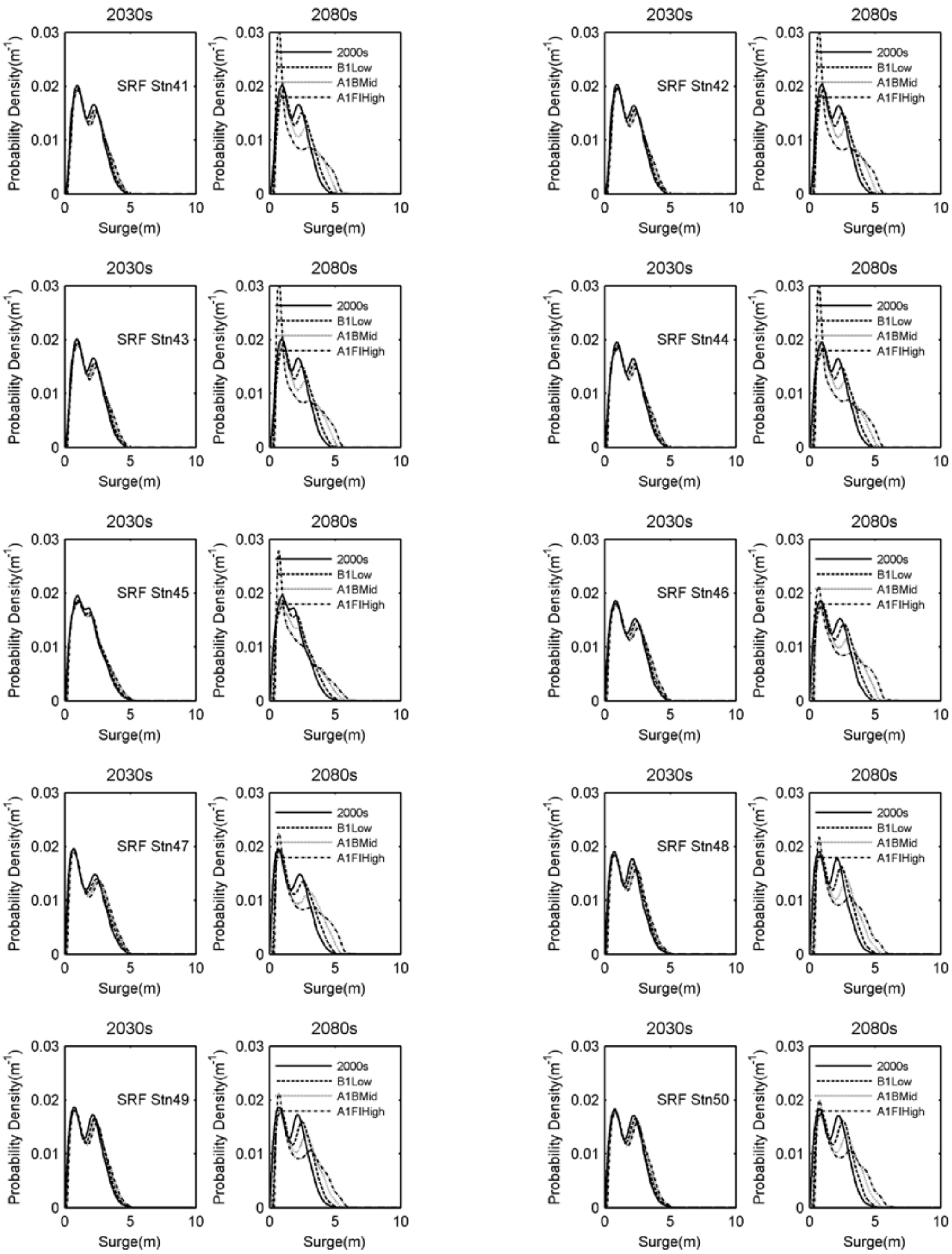


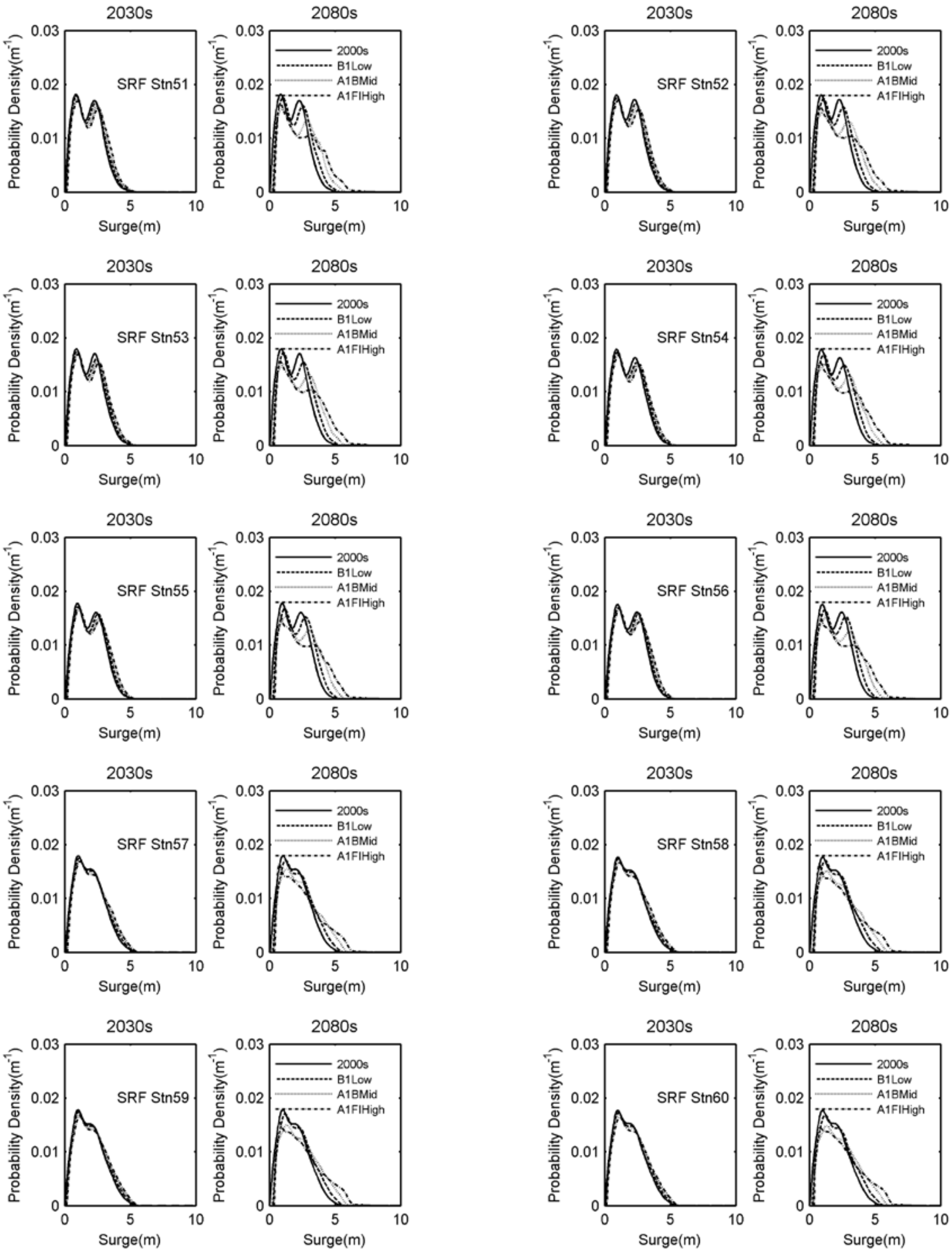


2030s
2080s
Probability Density(m^{-1})
Surge(m)
SRF Stn11
SRF Stn12
SRF Stn13
SRF Stn14
SRF Stn15
SRF Stn16
SRF Stn17
SRF Stn18
SRF Stn19
SRF Stn20
2000s
B1Low
A1BMid
A1FIHigh
0
0.01
0.02
0.03
0
5
10

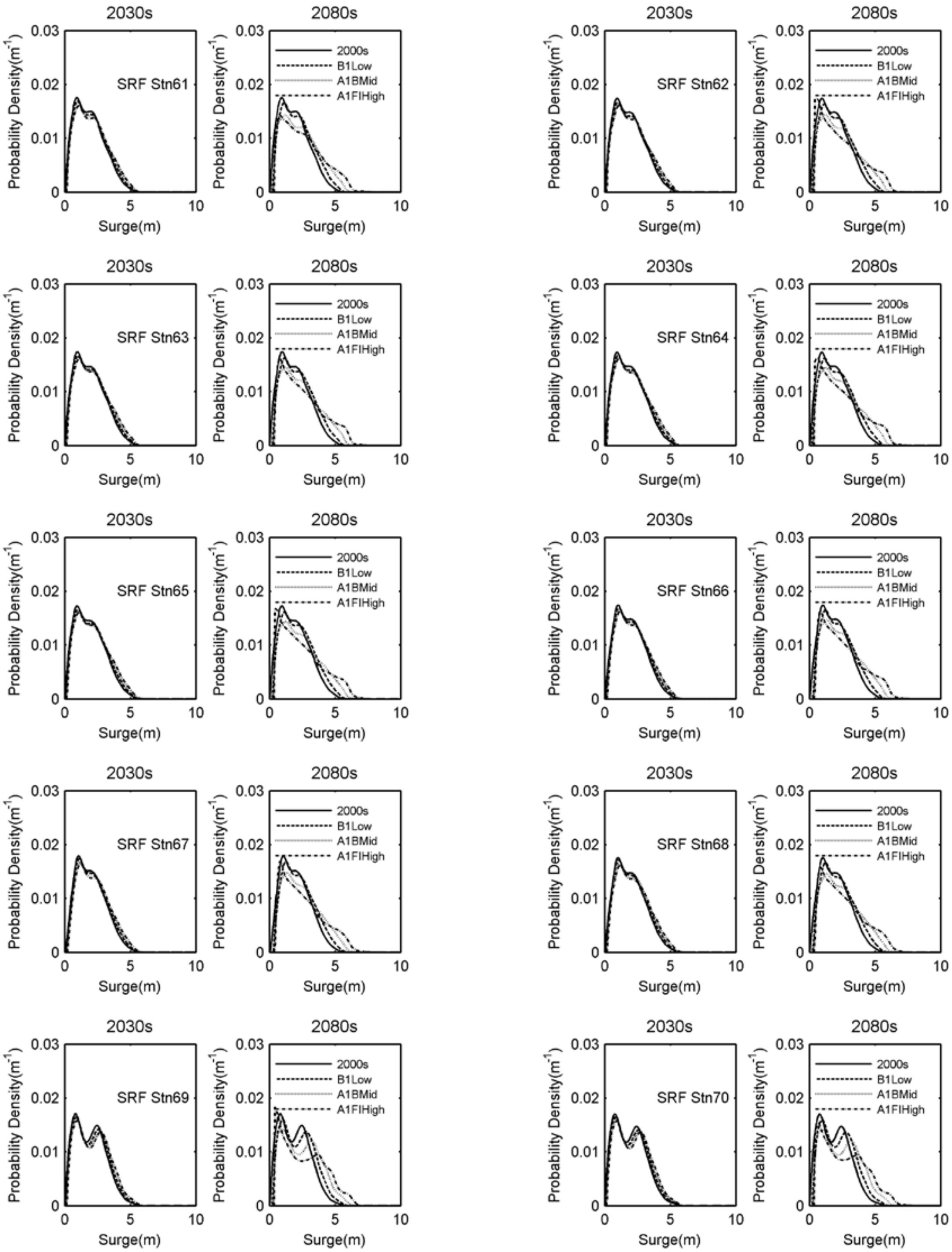

2030s
2080s
SRF Stn21
SRF Stn22
SRF Stn23
SRF Stn24
SRF Stn25
SRF Stn26
SRF Stn27
SRF Stn28
SRF Stn29
SRF Stn30
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)
0
0.01
0.02
0.03
5
10

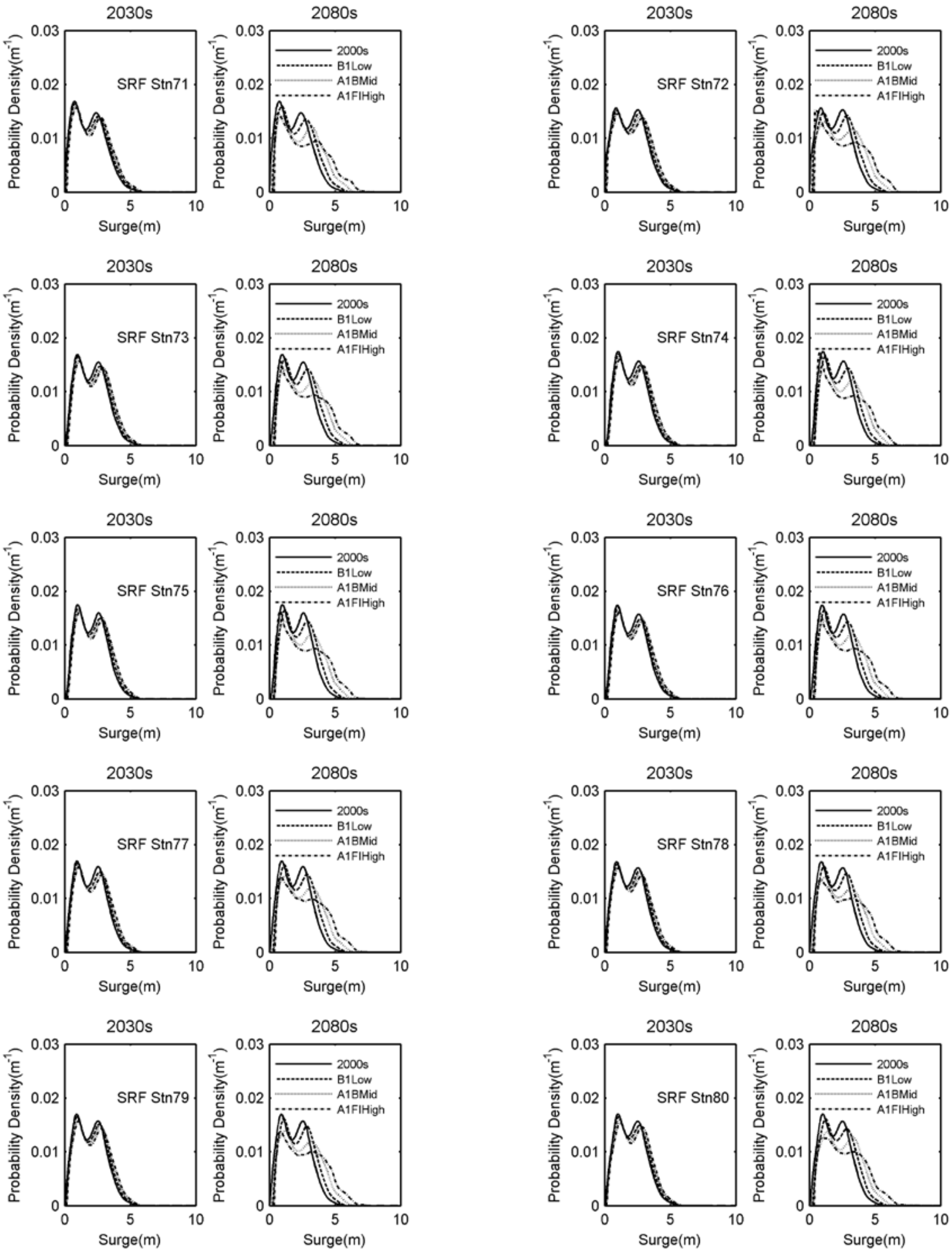

2030s
2080s
SRF Stn31
SRF Stn32
SRF Stn33
SRF Stn34
SRF Stn35
SRF Stn36
SRF Stn37
SRF Stn38
SRF Stn39
SRF Stn40
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)
0
0.01
0.02
0.03
5
10

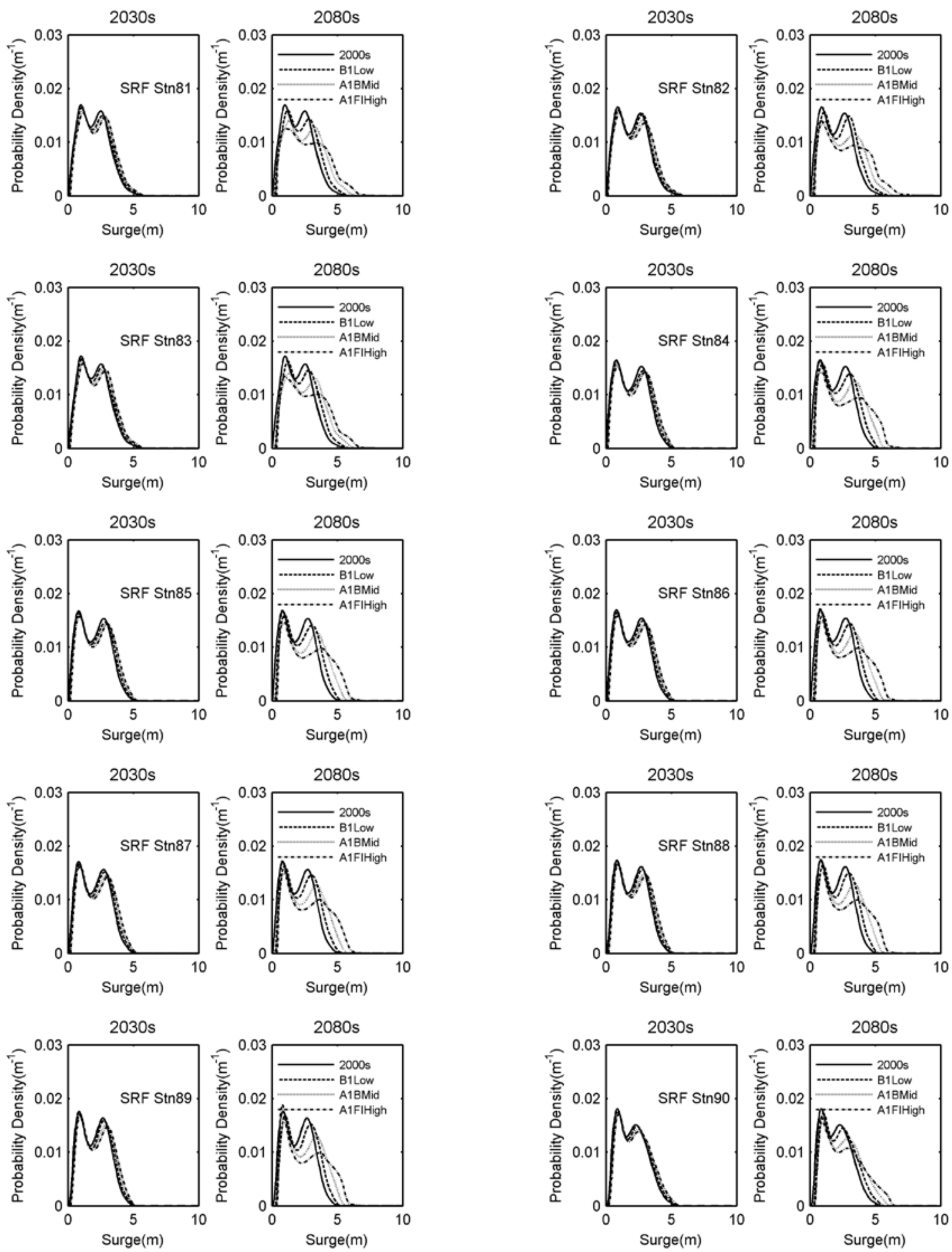

2030s
2080s
Probability Density(m^{-1})
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh
SRF Stn41
SRF Stn42
SRF Stn43
SRF Stn44
SRF Stn45
SRF Stn46
SRF Stn47
SRF Stn48
SRF Stn49
SRF Stn50

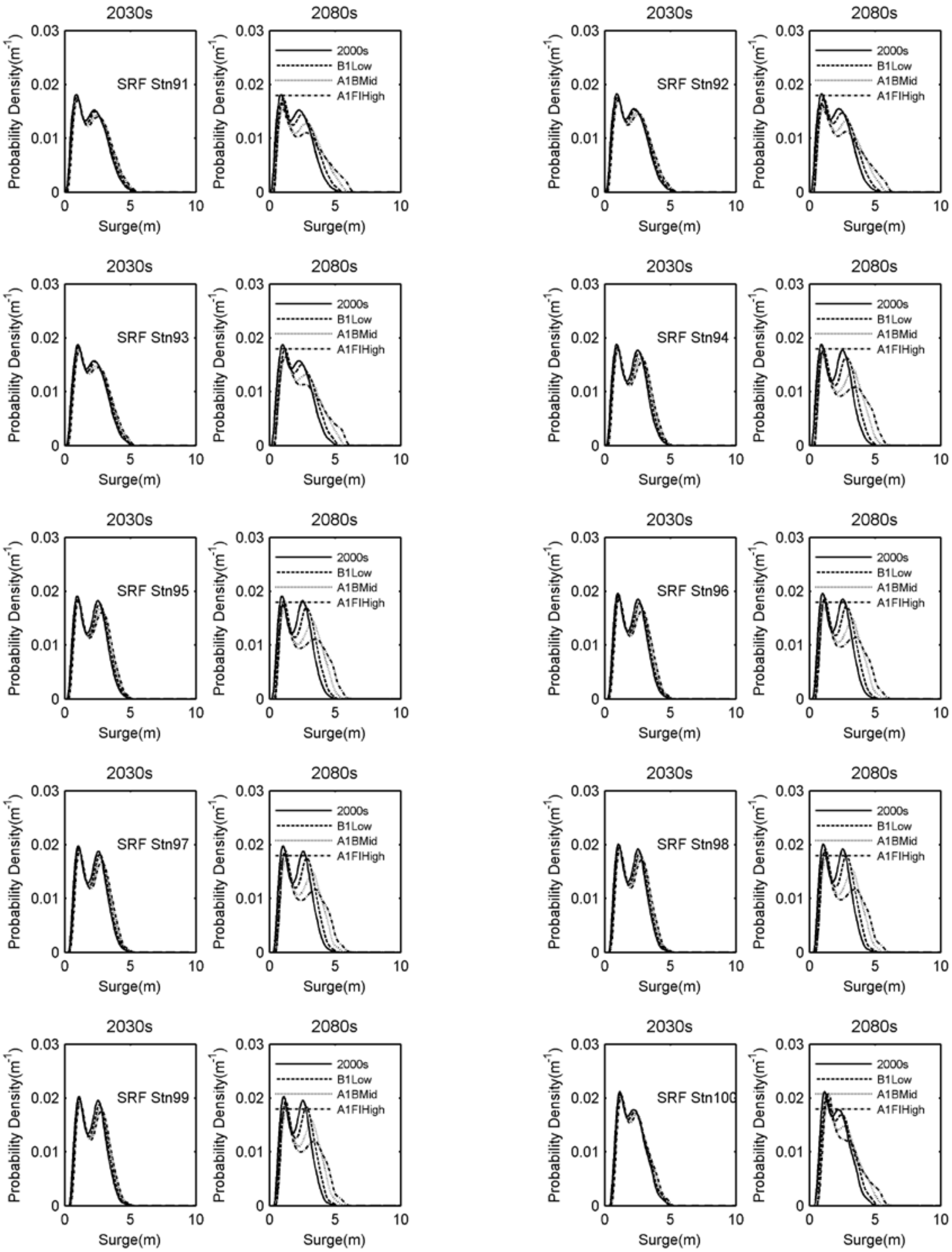


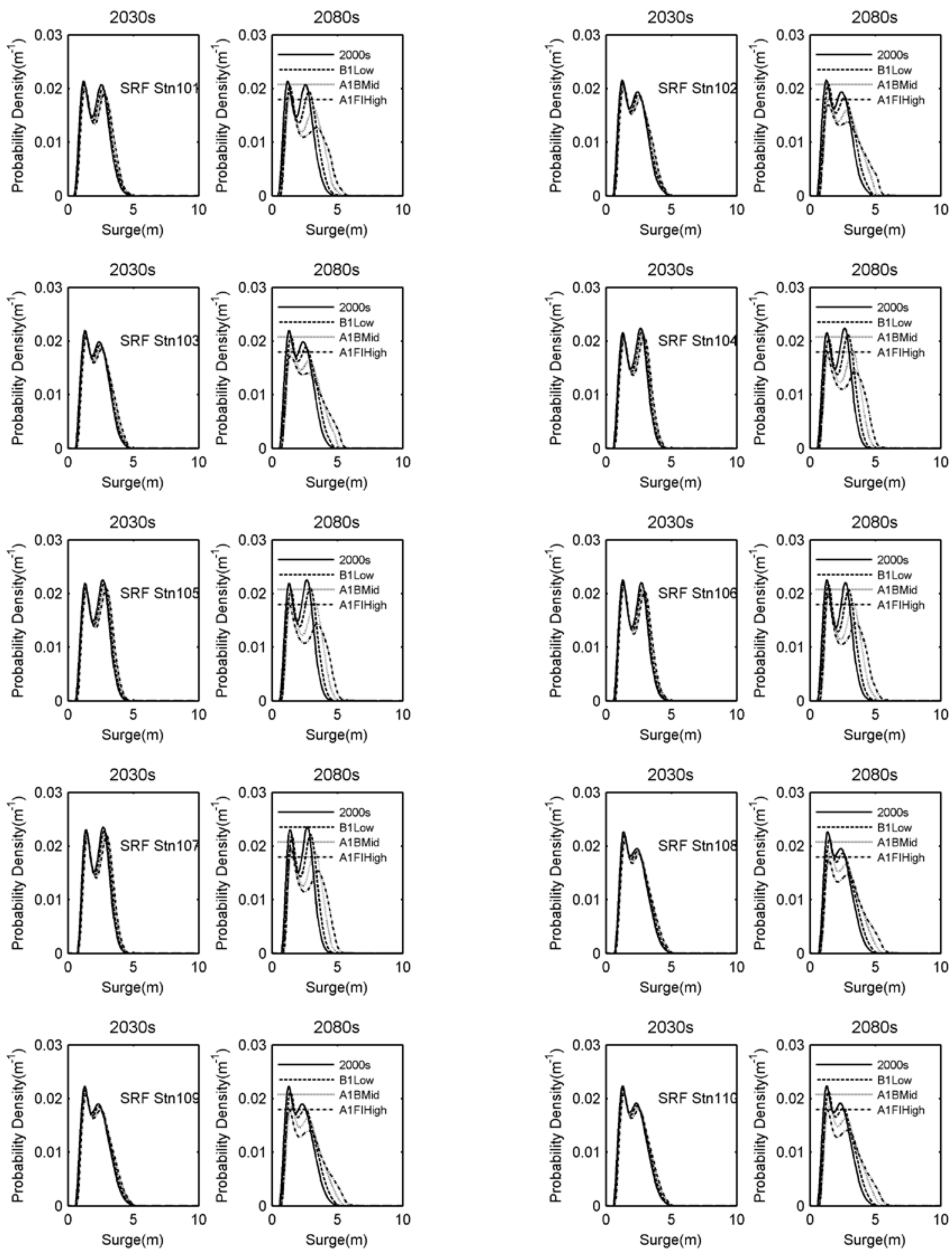
2030s
2080s
SRF Stn51
SRF Stn52
SRF Stn53
SRF Stn54
SRF Stn55
SRF Stn56
SRF Stn57
SRF Stn58
SRF Stn59
SRF Stn60
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)


2030s
2080s
SRF Stn61
SRF Stn62
SRF Stn63
SRF Stn64
SRF Stn65
SRF Stn66
SRF Stn67
SRF Stn68
SRF Stn69
SRF Stn70
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)
0
0.01
0.02
0.03
0
5
10

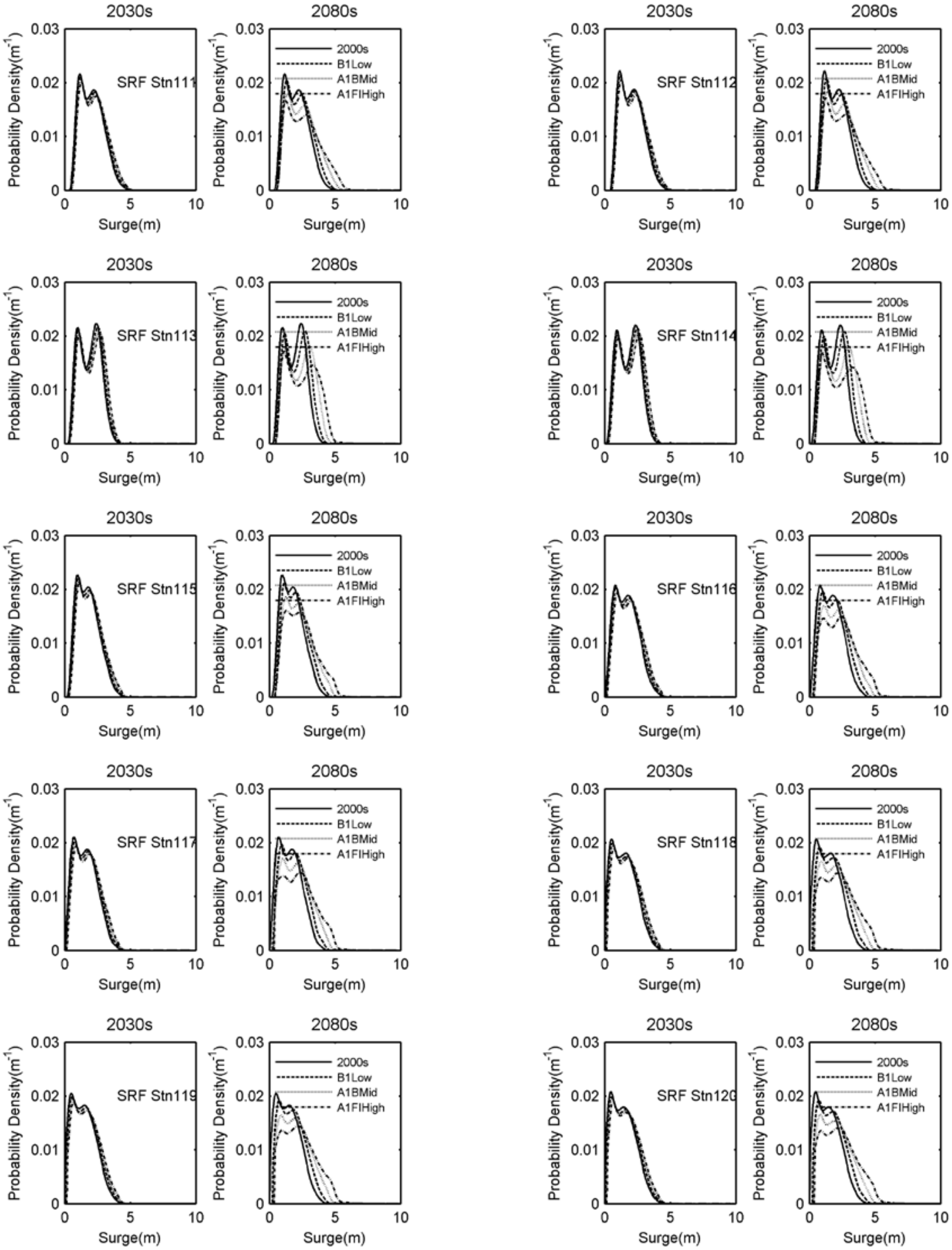

2030s
2080s
Probability Density(m^{-1})
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh
SRF Stn71
SRF Stn72
SRF Stn73
SRF Stn74
SRF Stn75
SRF Stn76
SRF Stn77
SRF Stn78
SRF Stn79
SRF Stn80

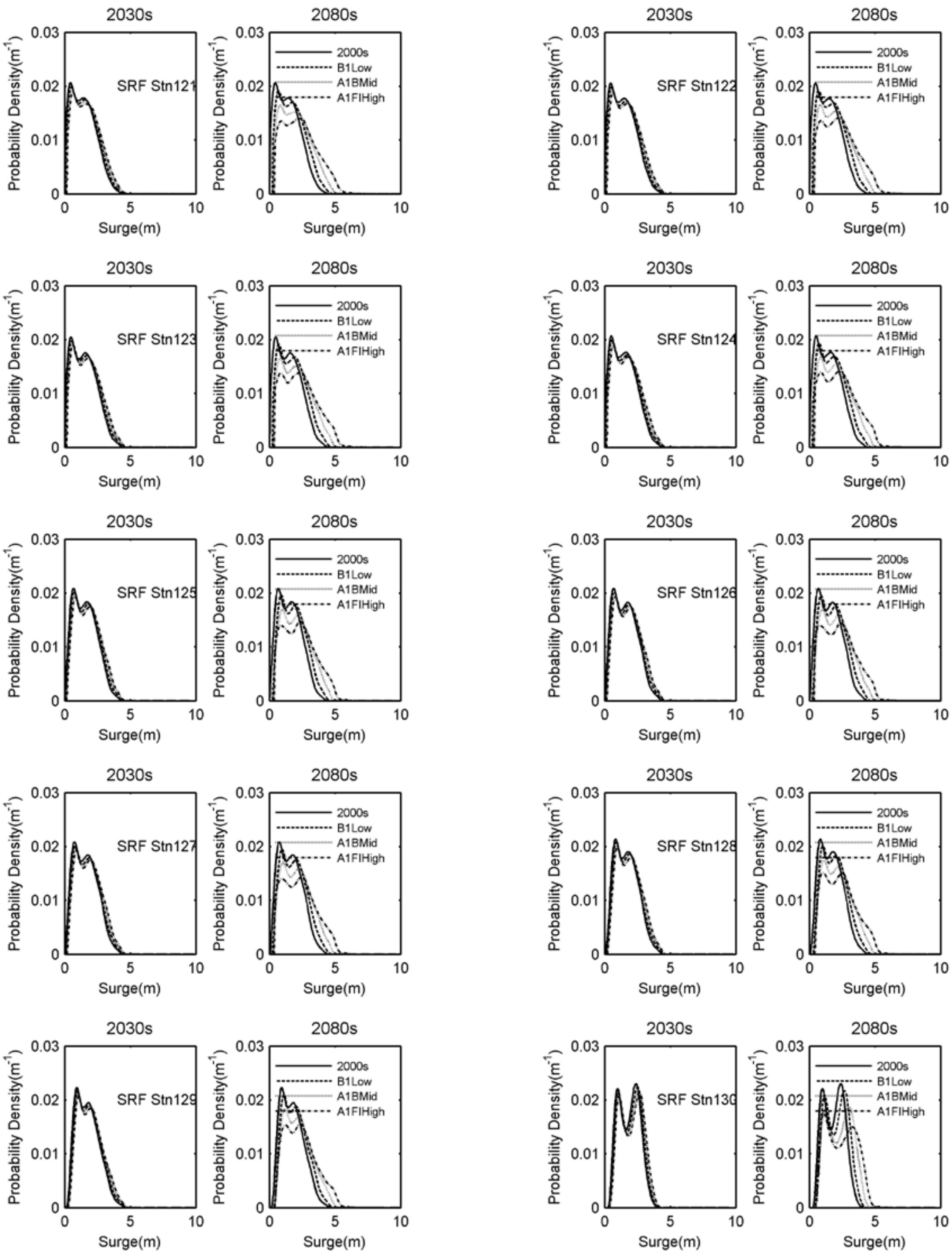

2030s
2080s
SRF Stn81
SRF Stn82
SRF Stn83
SRF Stn84
SRF Stn85
SRF Stn86
SRF Stn87
SRF Stn88
SRF Stn89
SRF Stn90
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)
0
0.01
0.02
0.03
0
5
10

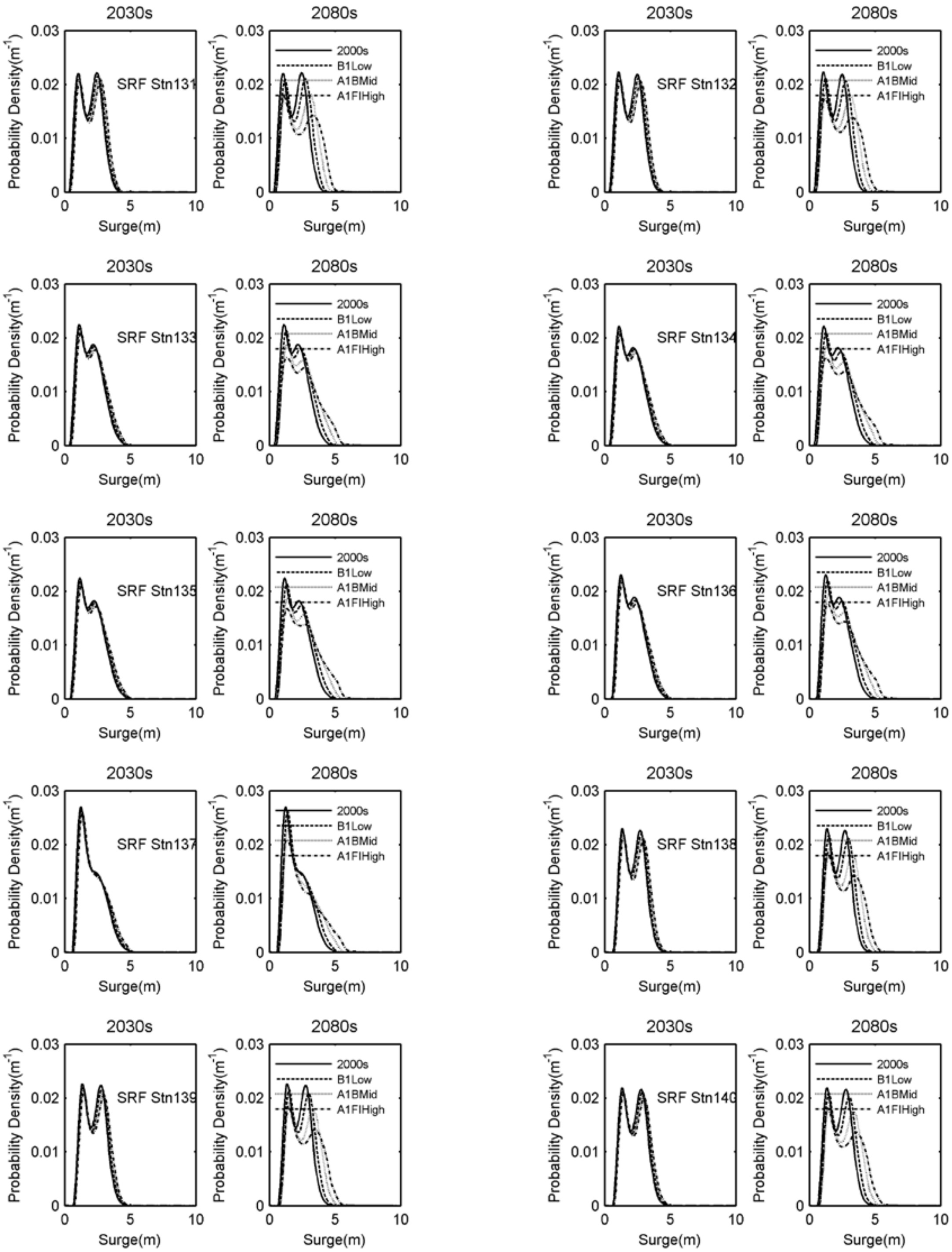

2030s
2080s
SRF Stn91
SRF Stn92
SRF Stn93
SRF Stn94
SRF Stn95
SRF Stn96
SRF Stn97
SRF Stn98
SRF Stn99
SRF Stn100
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)

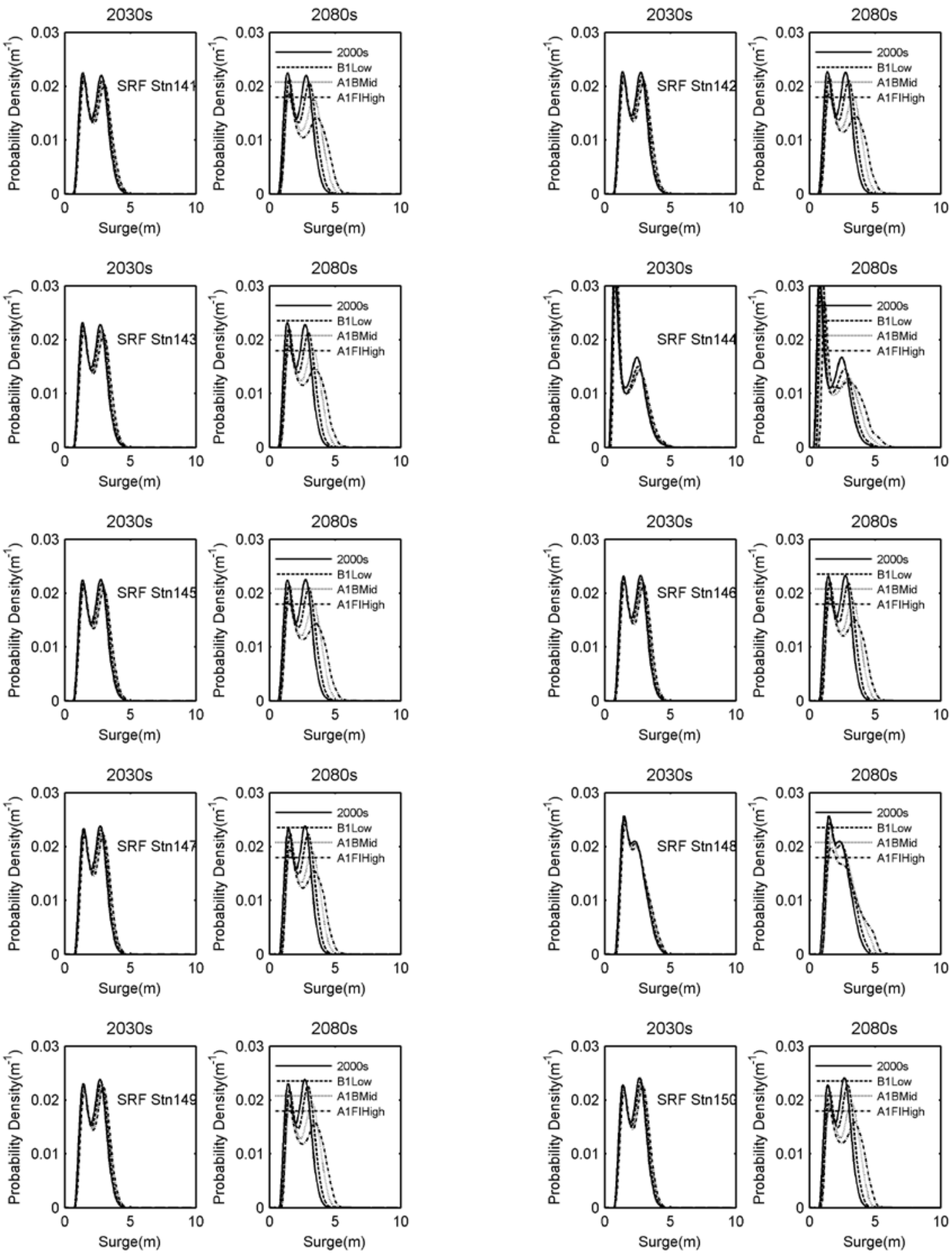


2030s
2080s
SRF Stn101
SRF Stn102
SRF Stn103
SRF Stn104
SRF Stn105
SRF Stn106
SRF Stn107
SRF Stn108
SRF Stn109
SRF Stn110
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)
0
0.01
0.02
0.03
5
10

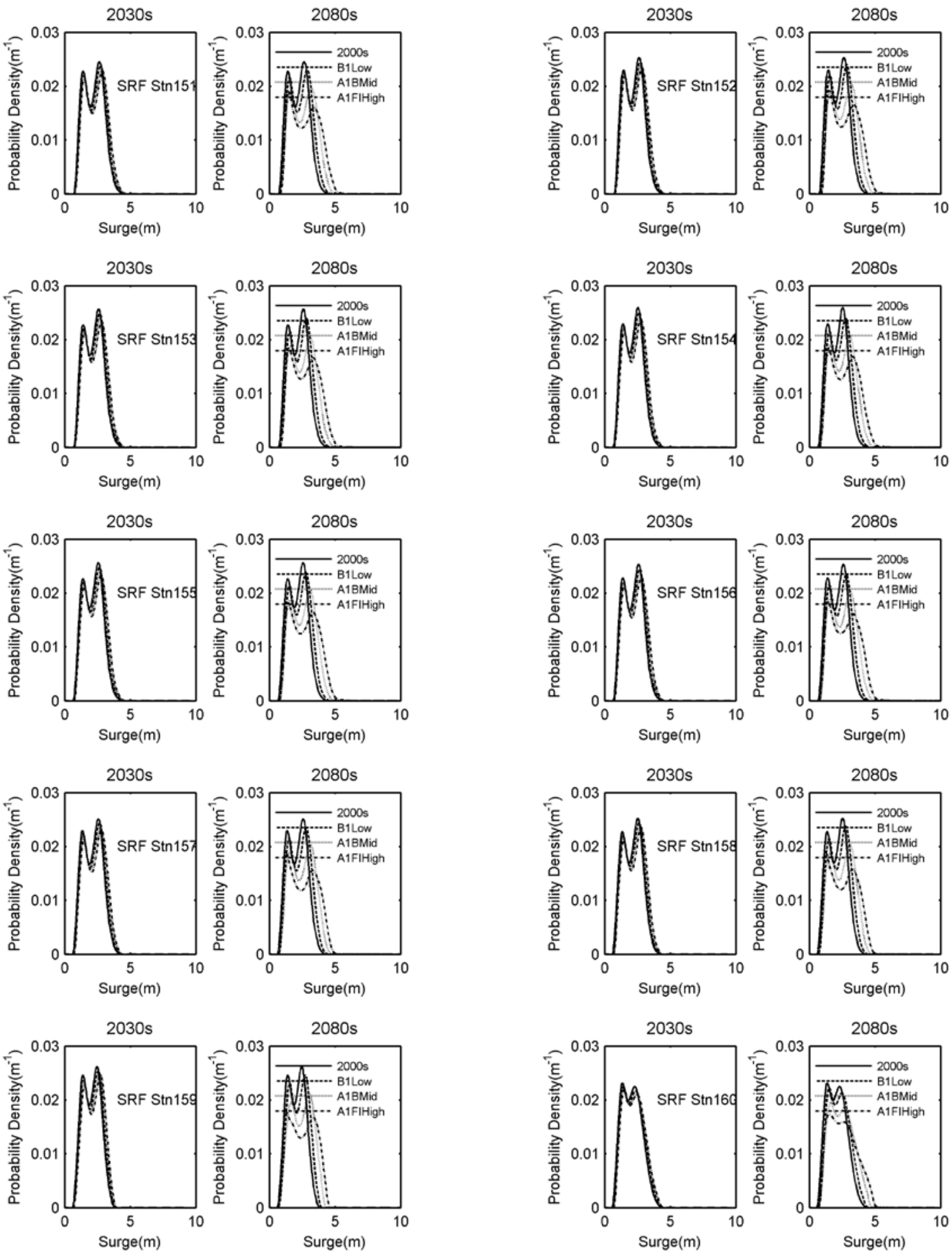

2030s
2080s
Probability Density(m^{-1})
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh
SRF Stn111
SRF Stn112
SRF Stn113
SRF Stn114
SRF Stn115
SRF Stn116
SRF Stn117
SRF Stn118
SRF Stn119
SRF Stn120

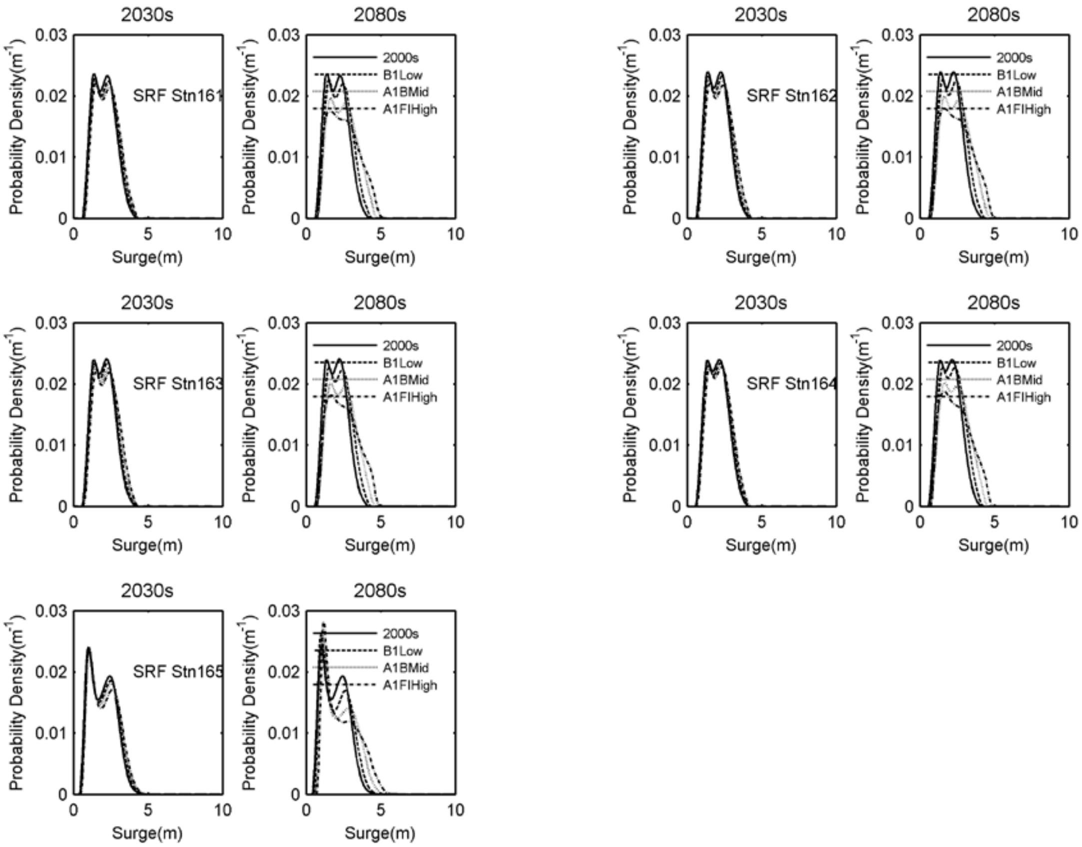

2030s
2080s
Probability Density(m^{-1})
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh
SRF Stn121
SRF Stn122
SRF Stn123
SRF Stn124
SRF Stn125
SRF Stn126
SRF Stn127
SRF Stn128
SRF Stn129
SRF Stn130


2030s
2080s
Probability Density(m-1)
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh
SRF Stn131
SRF Stn132
SRF Stn133
SRF Stn134
SRF Stn135
SRF Stn136
SRF Stn137
SRF Stn138
SRF Stn139
SRF Stn140

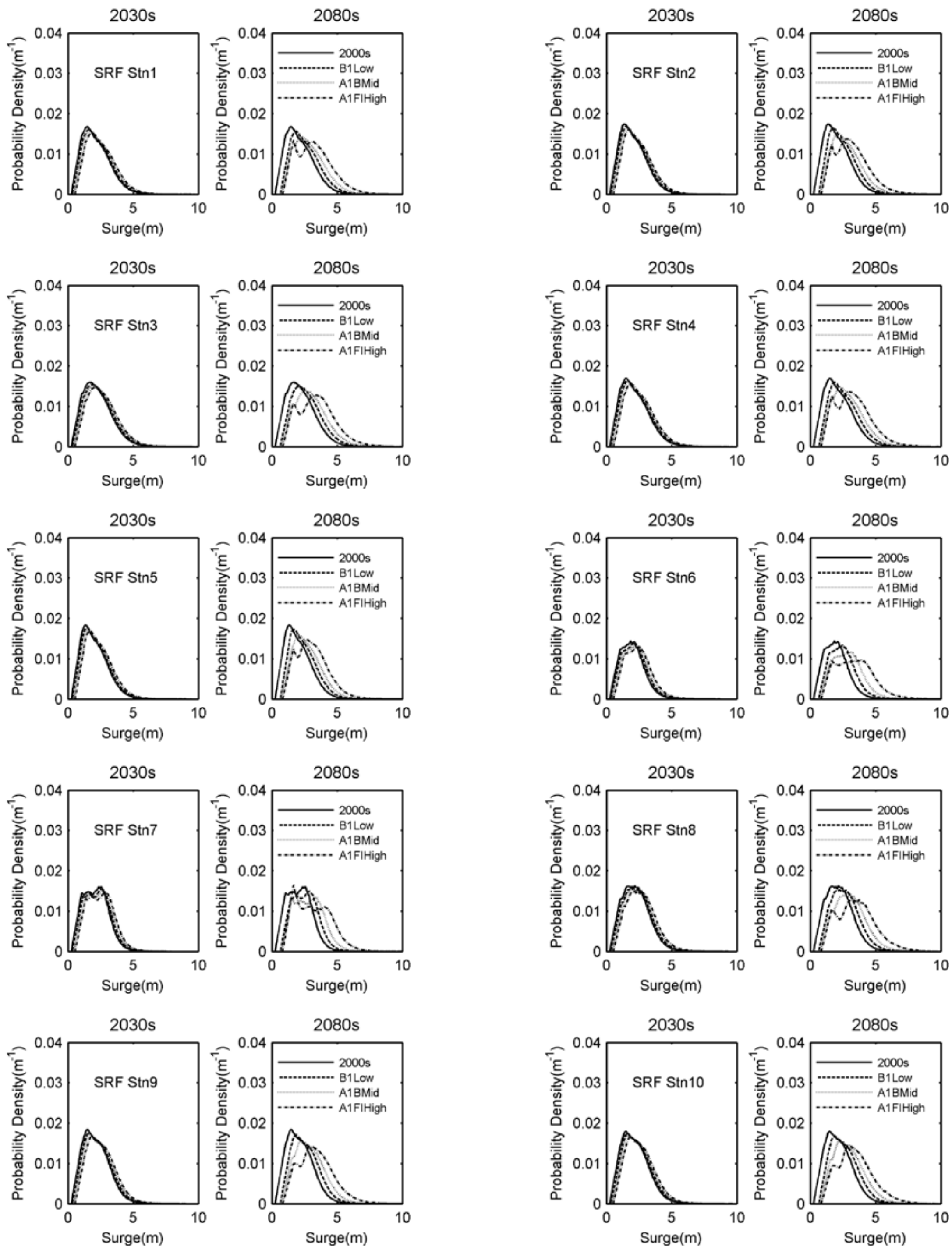


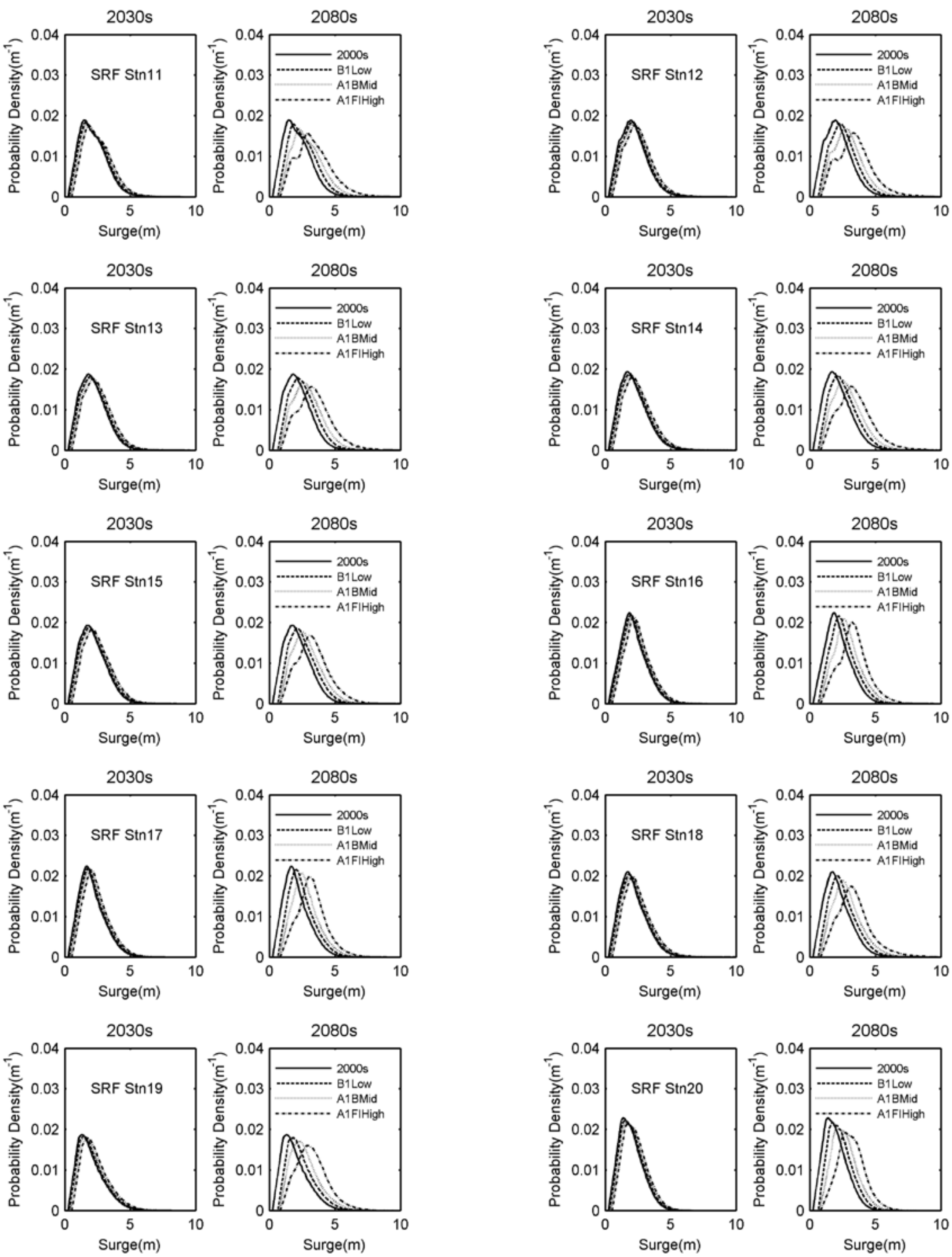
2030s
2080s
SRF Stn141
SRF Stn142
SRF Stn143
SRF Stn144
SRF Stn145
SRF Stn146
SRF Stn147
SRF Stn148
SRF Stn149
SRF Stn150
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^-1)
Surge(m)

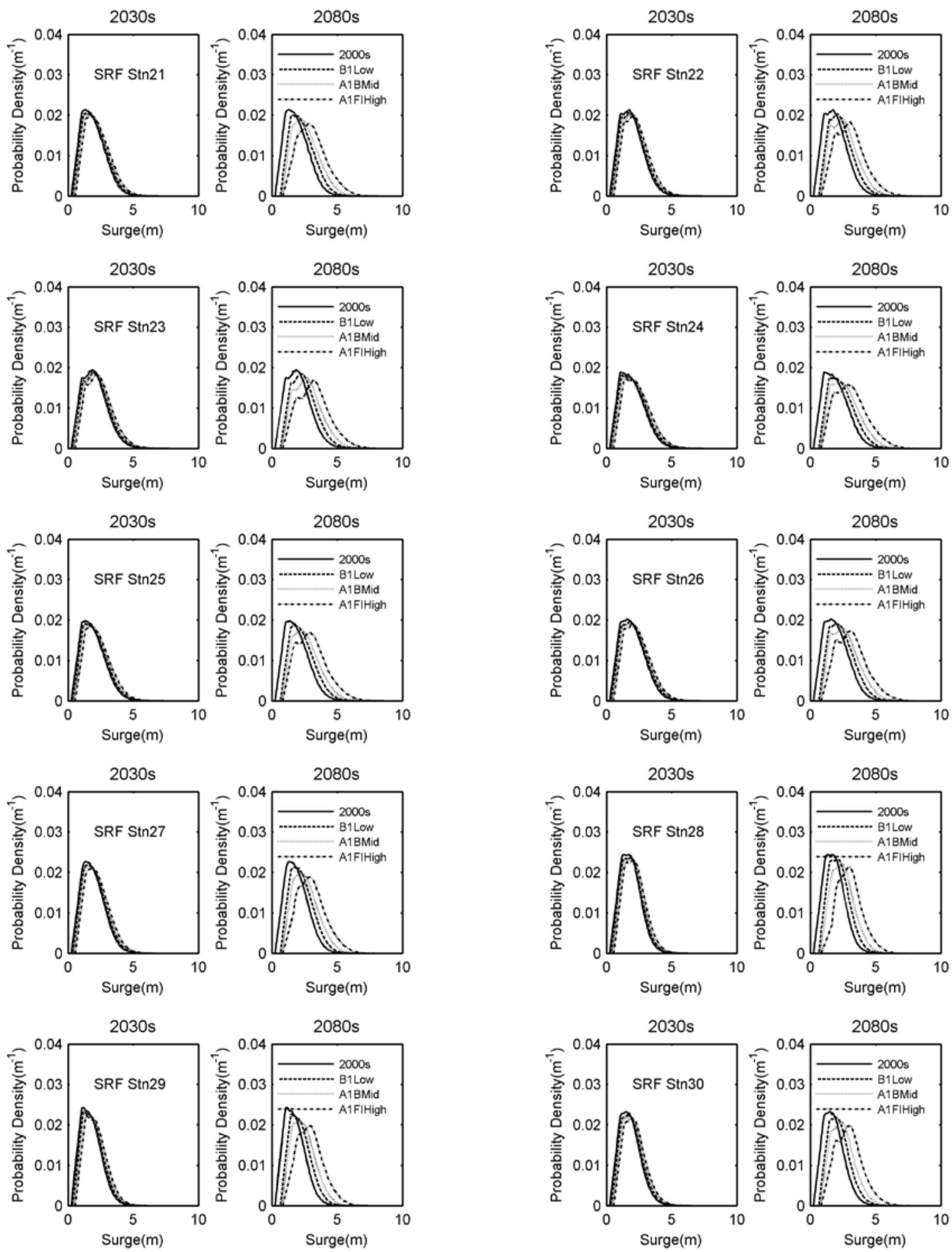

2030s
2080s
Probability Density(m^{-1})
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh
SRF Stn151
SRF Stn152
SRF Stn153
SRF Stn154
SRF Stn155
SRF Stn156
SRF Stn157
SRF Stn158
SRF Stn159
SRF Stn160


2030s
2080s
SRF Stn161
SRF Stn162
SRF Stn163
SRF Stn164
SRF Stn165
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)
0
0.01
0.02
0.03
5
10

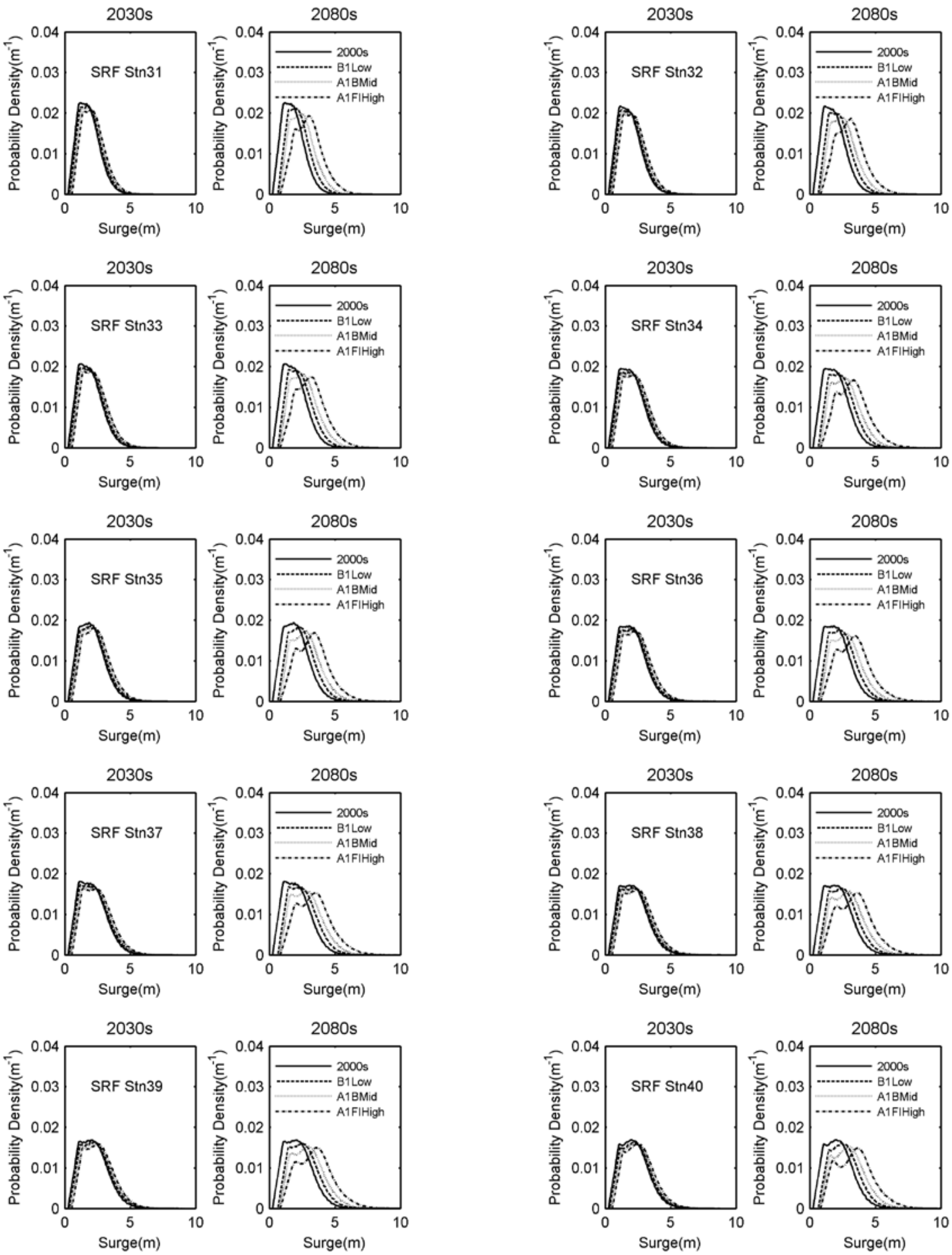
F.3. Panama, FL

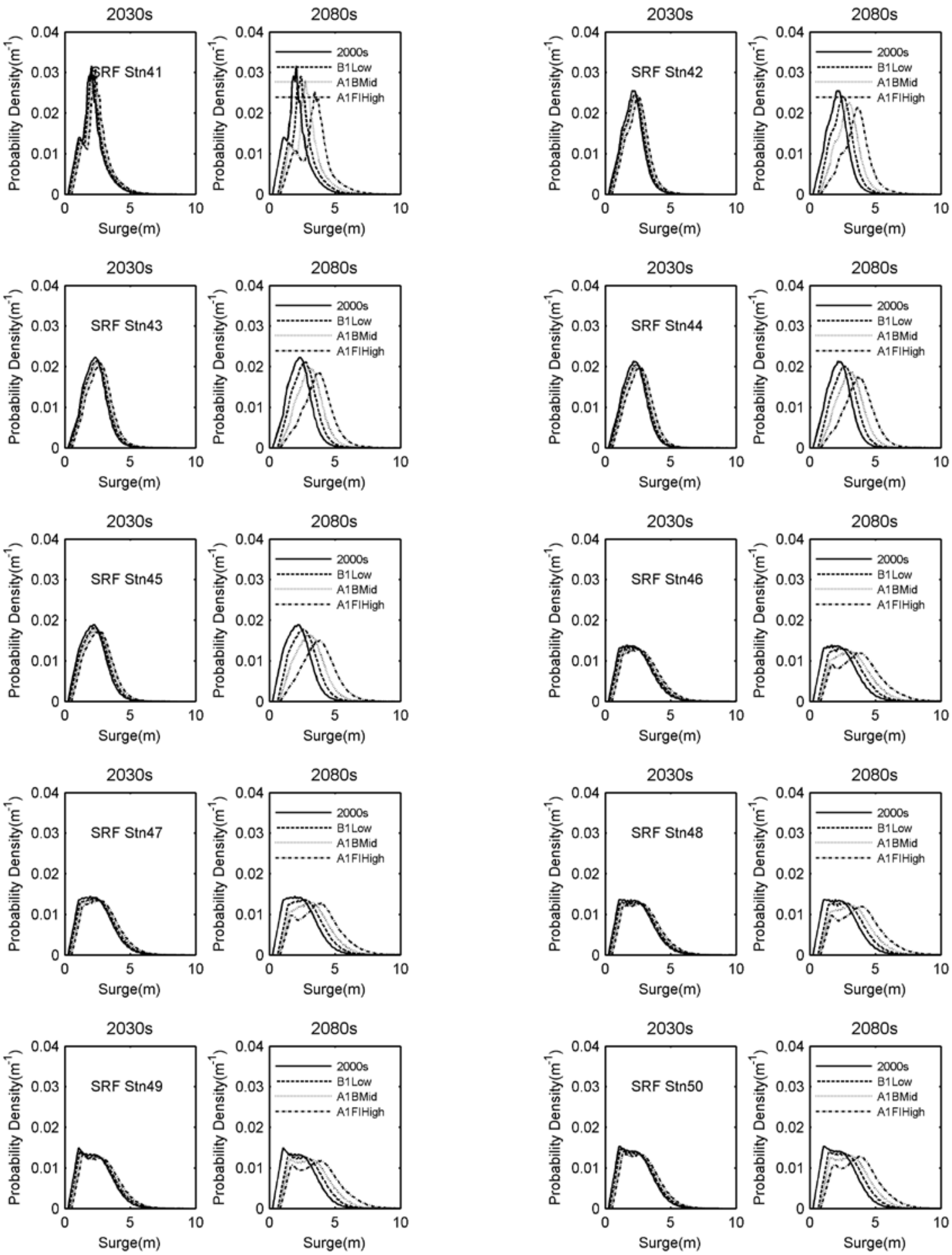

2030s
2080s
SRF Stn1
SRF Stn2
SRF Stn3
SRF Stn4
SRF Stn5
SRF Stn6
SRF Stn7
SRF Stn8
SRF Stn9
SRF Stn10
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m⁻¹)
Surge(m)

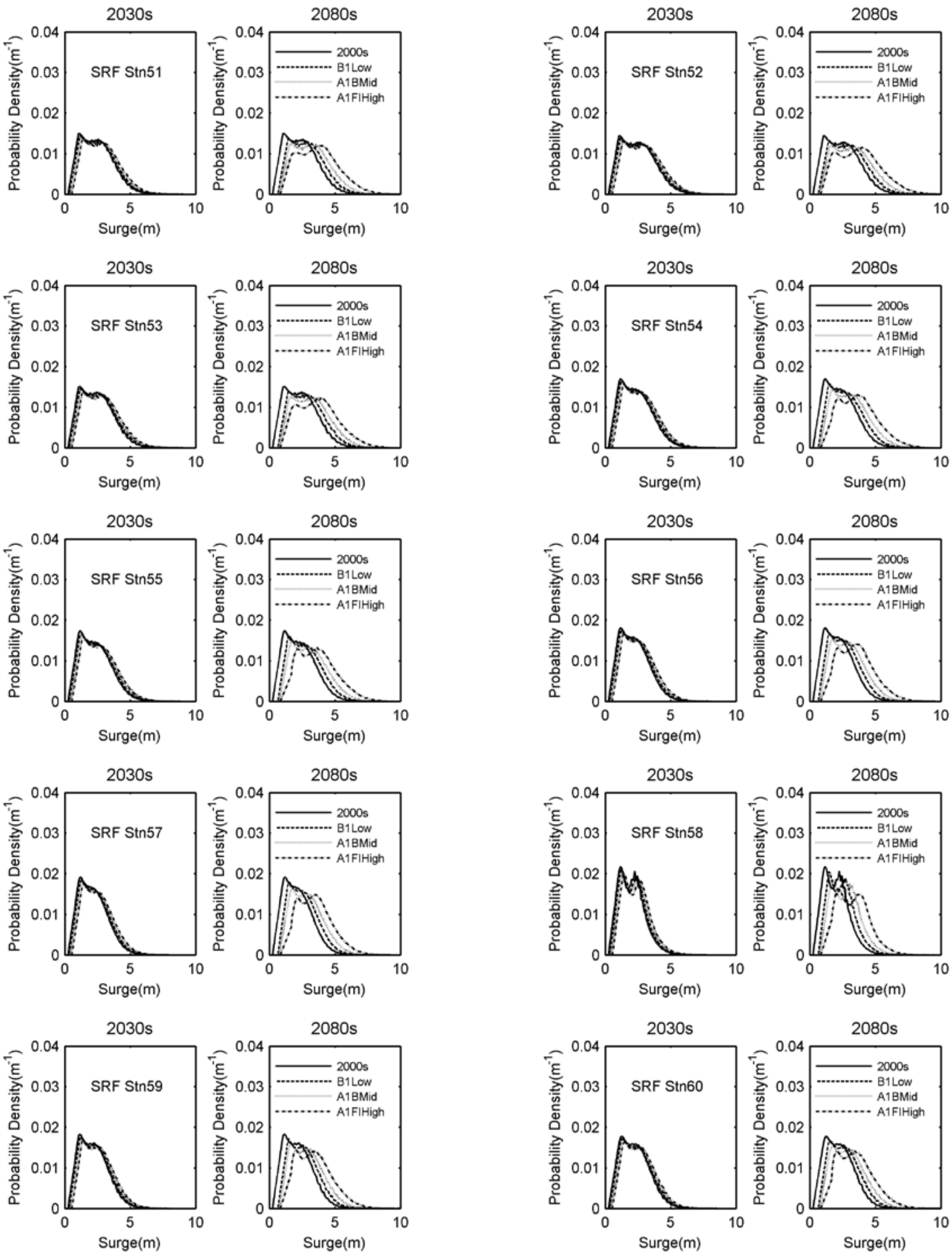

2030s
2080s
SRF Stn11
SRF Stn12
SRF Stn13
SRF Stn14
SRF Stn15
SRF Stn16
SRF Stn17
SRF Stn18
SRF Stn19
SRF Stn20
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)
0
0.01
0.02
0.03
0.04
5
10

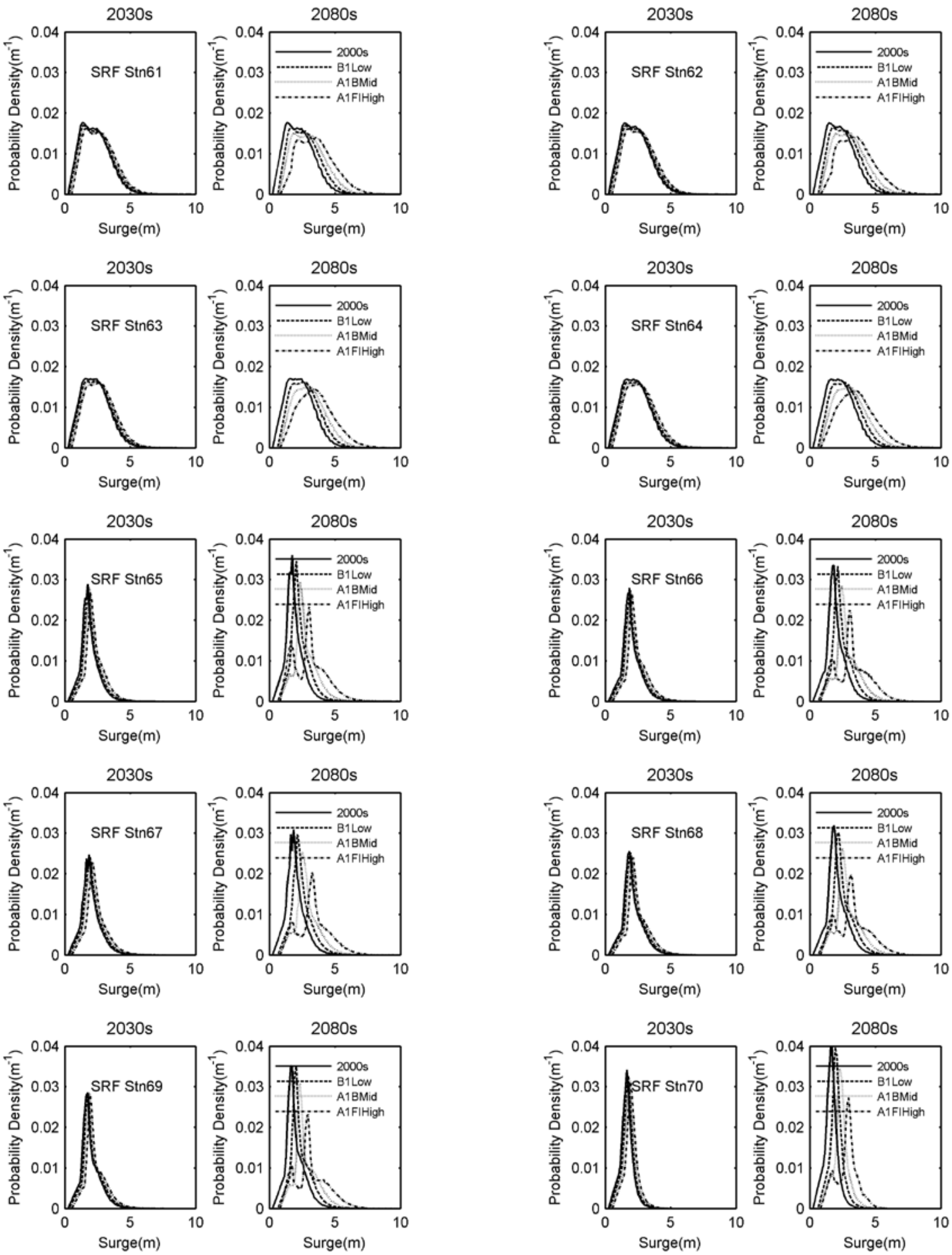


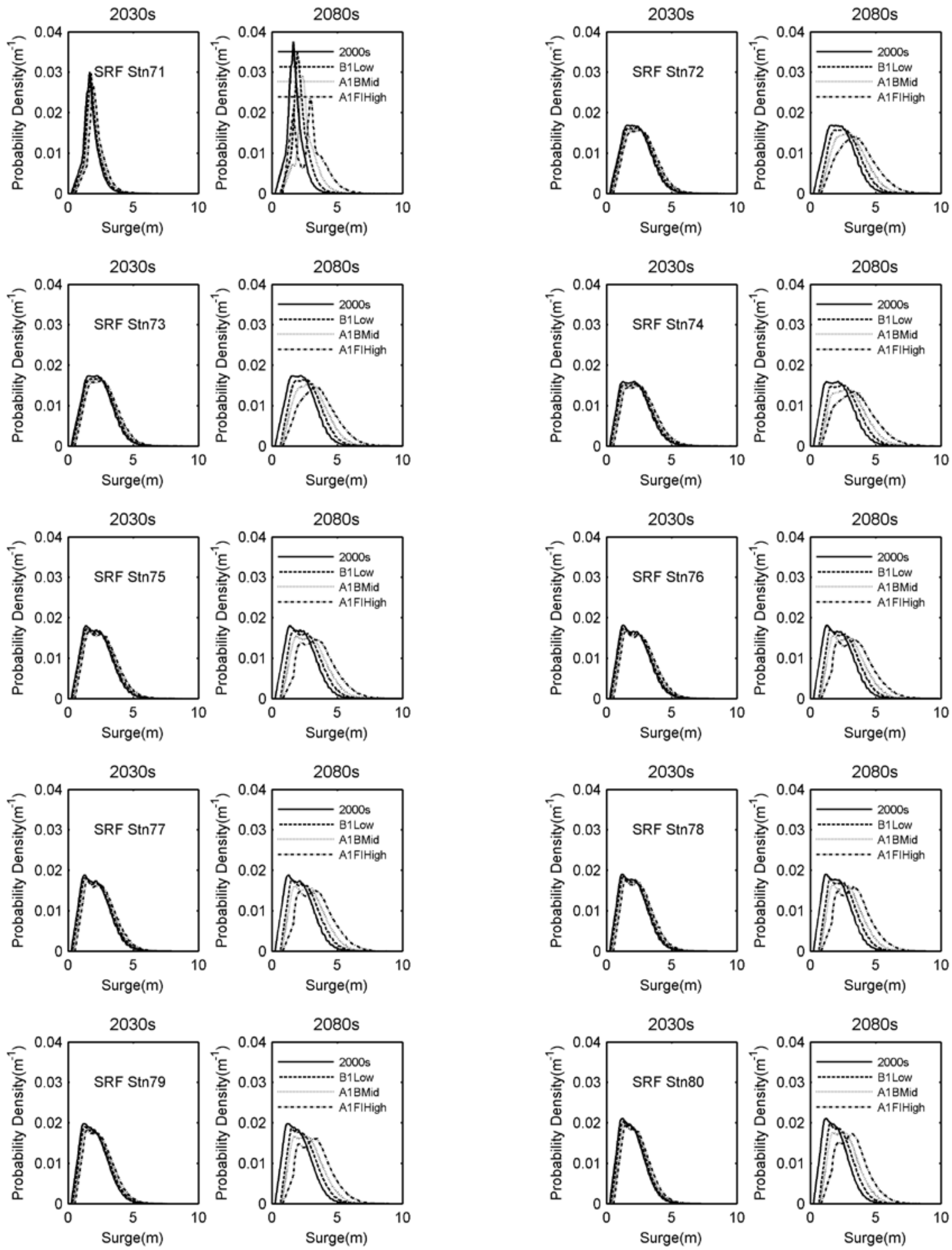
2030s
2080s
SRF Stn21
SRF Stn22
SRF Stn23
SRF Stn24
SRF Stn25
SRF Stn26
SRF Stn27
SRF Stn28
SRF Stn29
SRF Stn30
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)

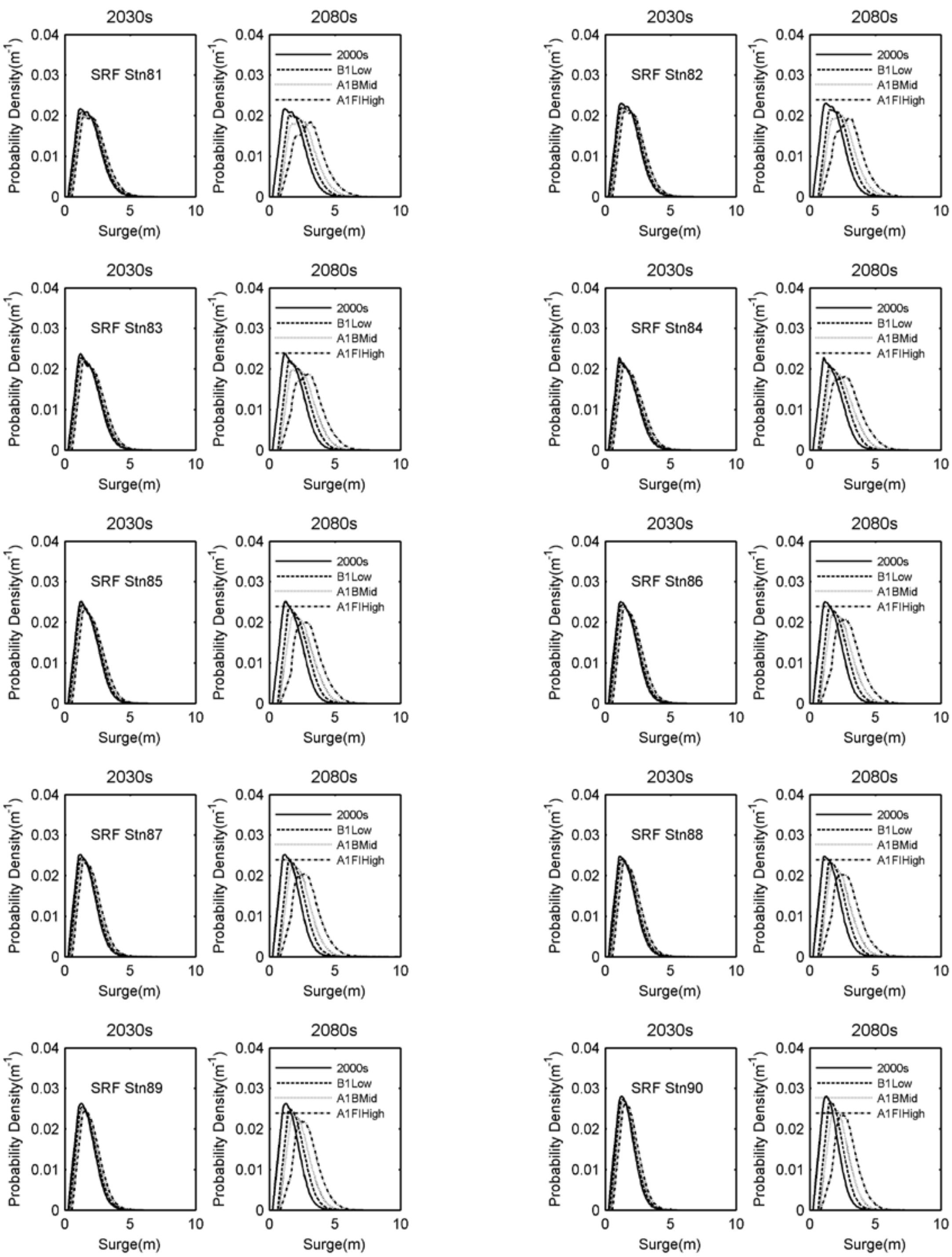

2030s
2080s
SRF Stn31
SRF Stn32
SRF Stn33
SRF Stn34
SRF Stn35
SRF Stn36
SRF Stn37
SRF Stn38
SRF Stn39
SRF Stn40
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)

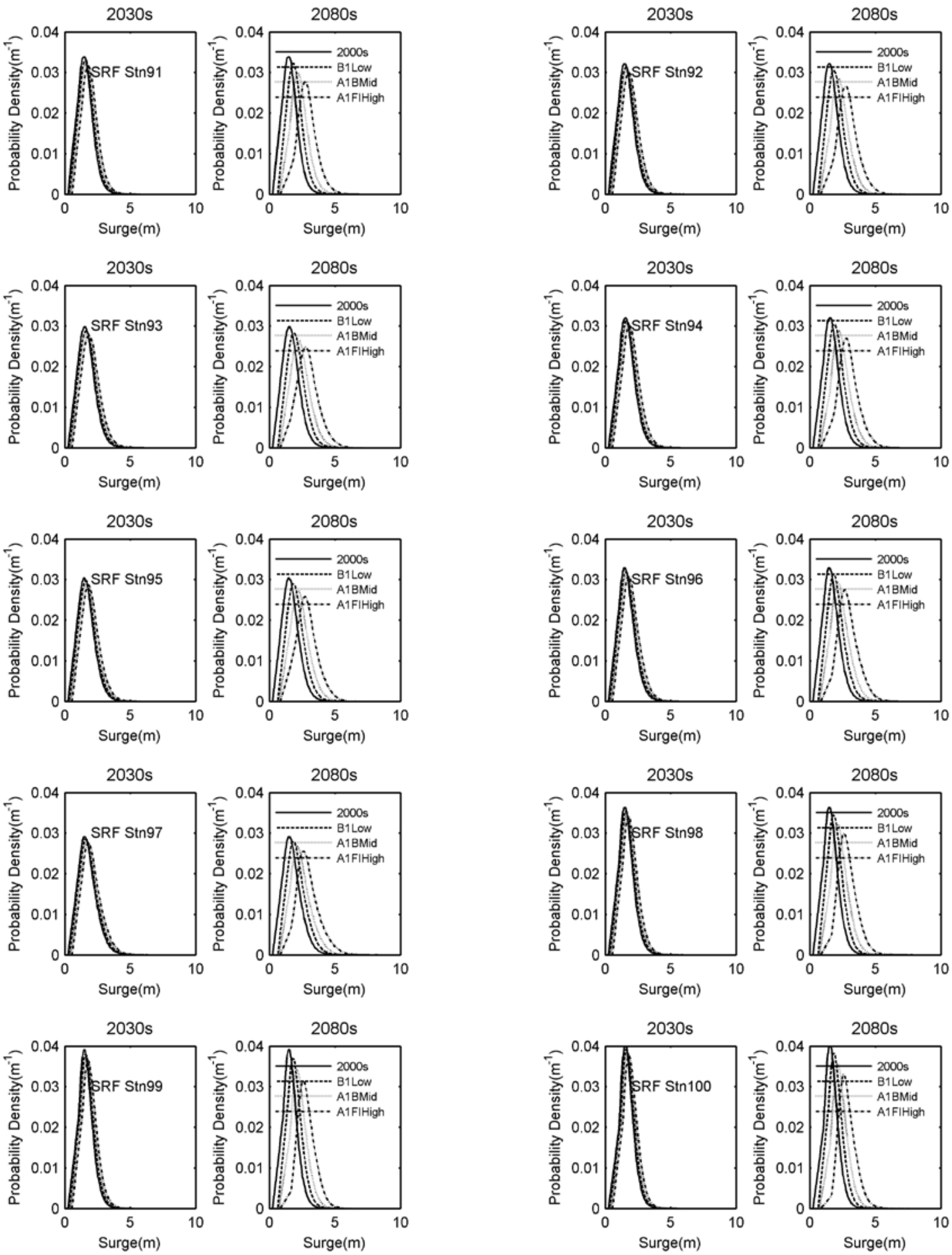

2030s
2080s
SRF Stn41
SRF Stn42
SRF Stn43
SRF Stn44
SRF Stn45
SRF Stn46
SRF Stn47
SRF Stn48
SRF Stn49
SRF Stn50
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)

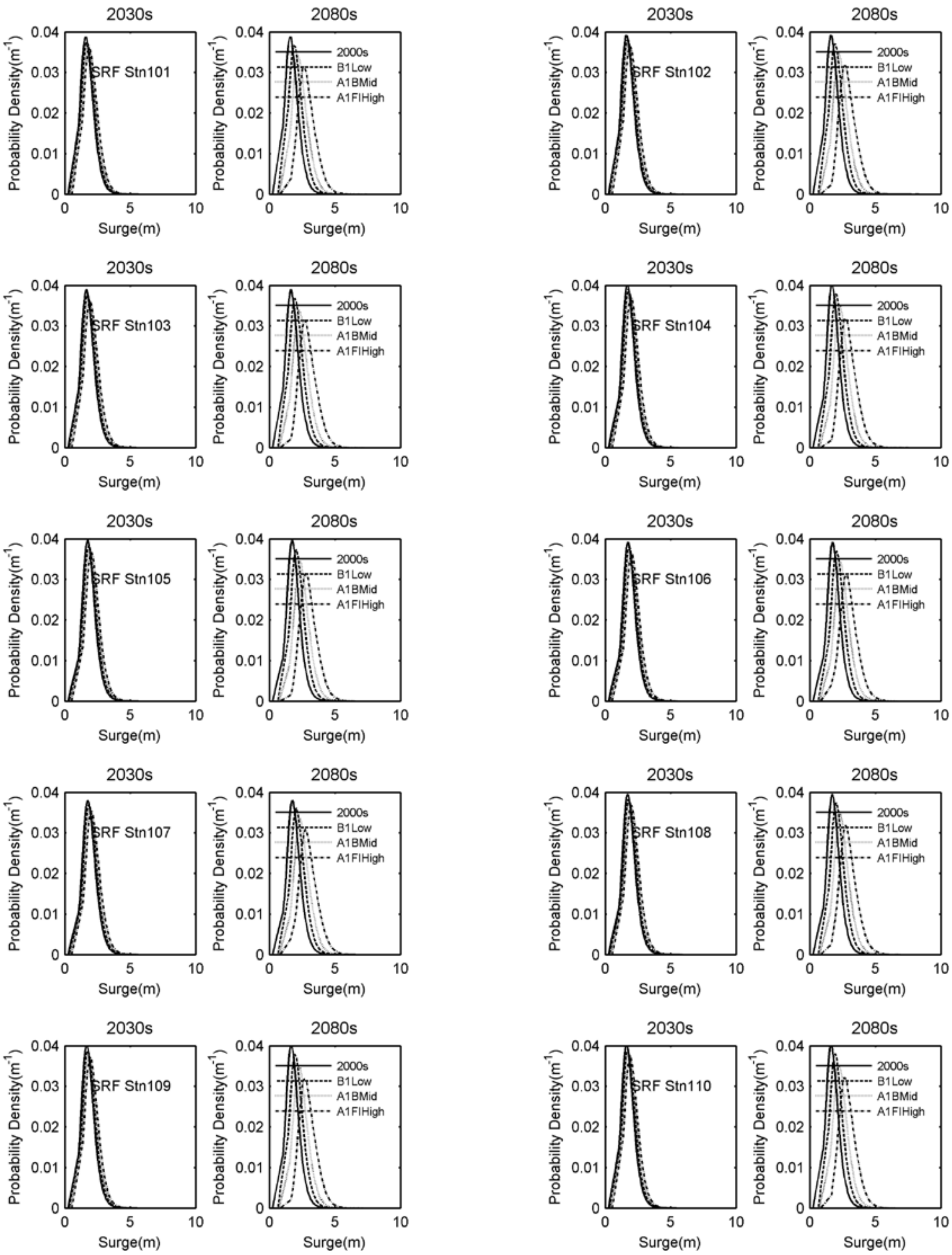

2030s
2080s
SRF Stn51
SRF Stn52
SRF Stn53
SRF Stn54
SRF Stn55
SRF Stn56
SRF Stn57
SRF Stn58
SRF Stn59
SRF Stn60
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)
0
0.01
0.02
0.03
0.04
5
10

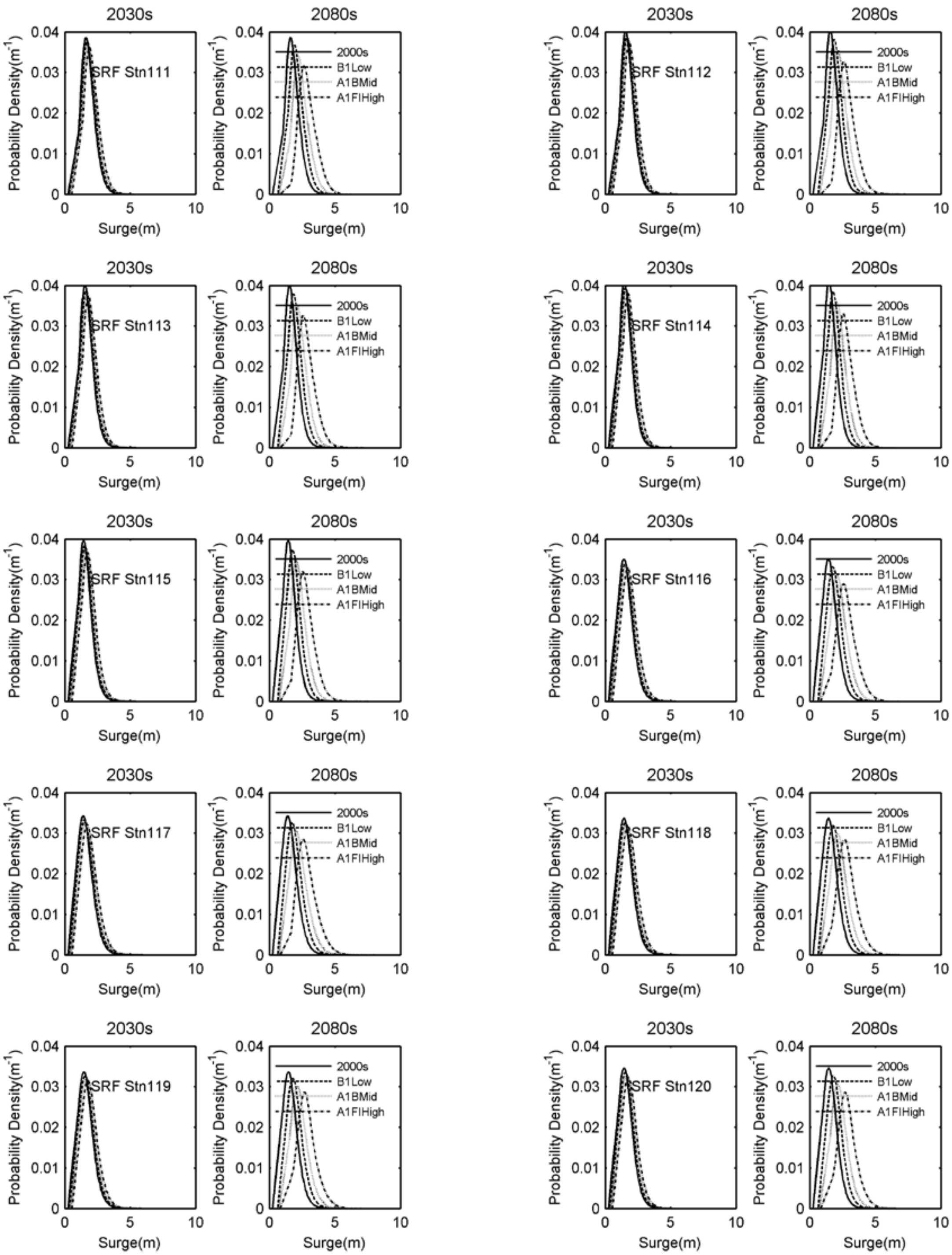

SRF Stn61
SRF Stn62
SRF Stn63
SRF Stn64
SRF Stn65
SRF Stn66
SRF Stn67
SRF Stn68
SRF Stn69
SRF Stn70
2030s
2080s
Probability Density(m^{-1})
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh

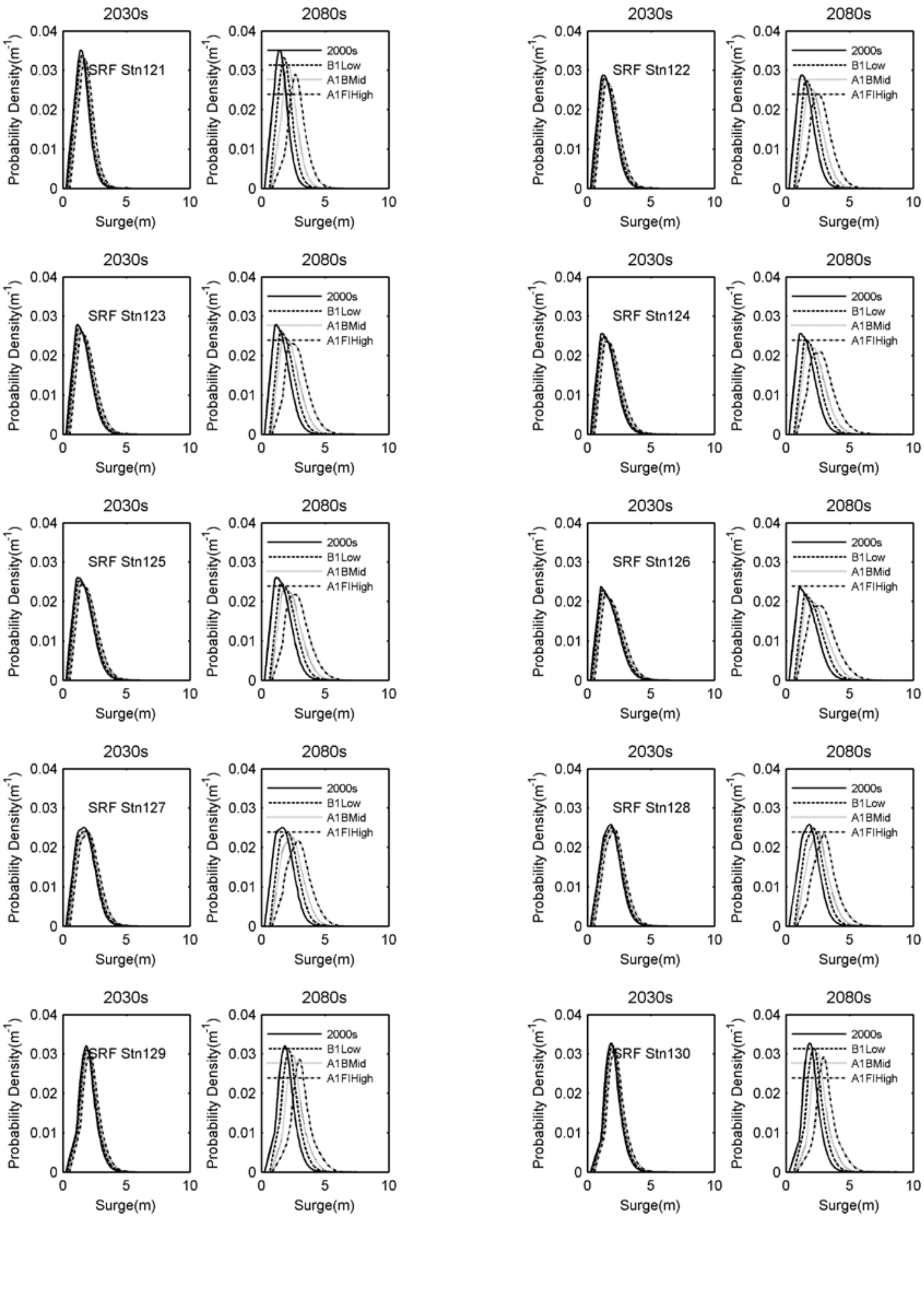

2030s
2080s
SRF Stn71
SRF Stn72
SRF Stn73
SRF Stn74
SRF Stn75
SRF Stn76
SRF Stn77
SRF Stn78
SRF Stn79
SRF Stn80
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)

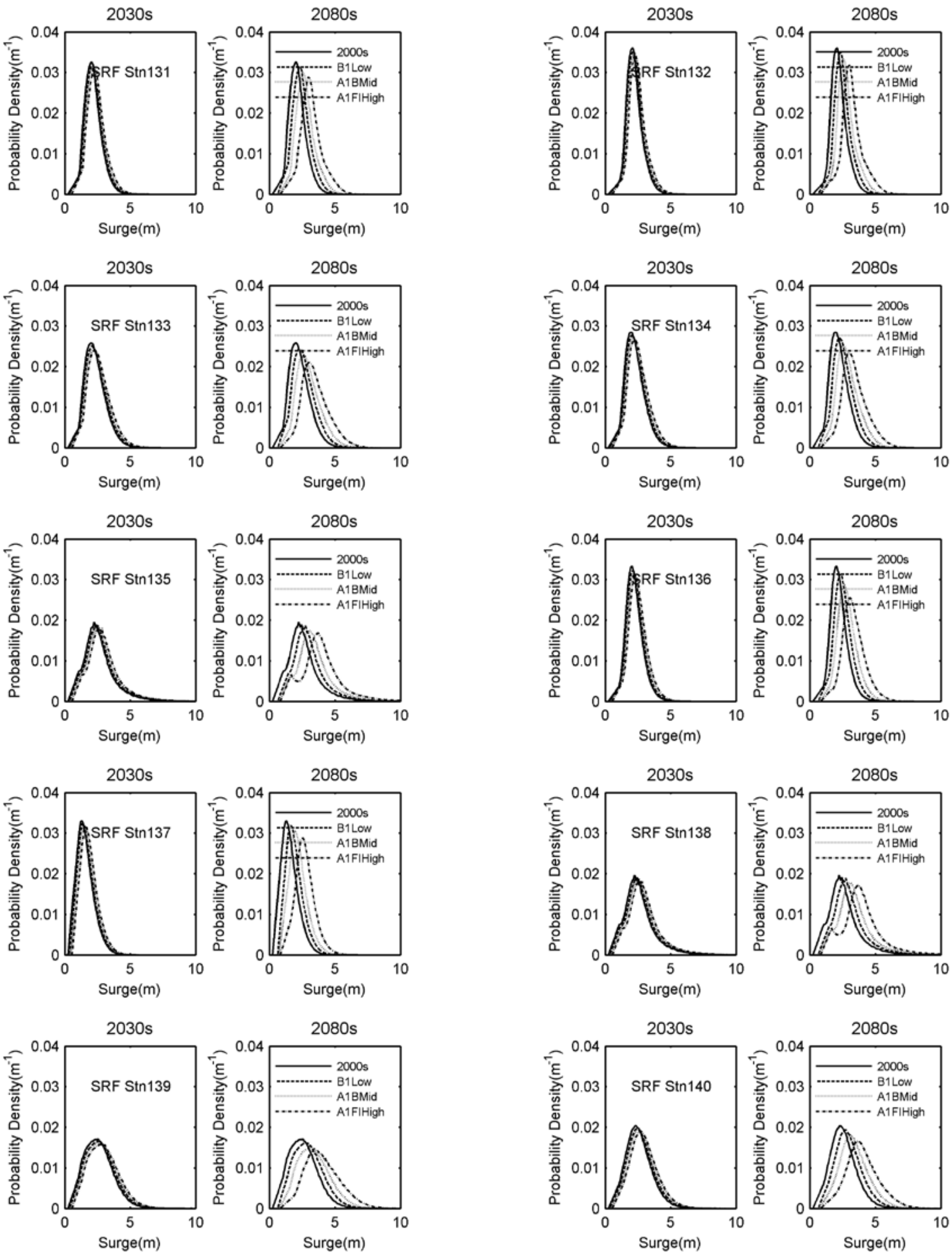

2030s
2080s
SRF Stn81
SRF Stn82
SRF Stn83
SRF Stn84
SRF Stn85
SRF Stn86
SRF Stn87
SRF Stn88
SRF Stn89
SRF Stn90
Probability Density(m^{-1})
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh

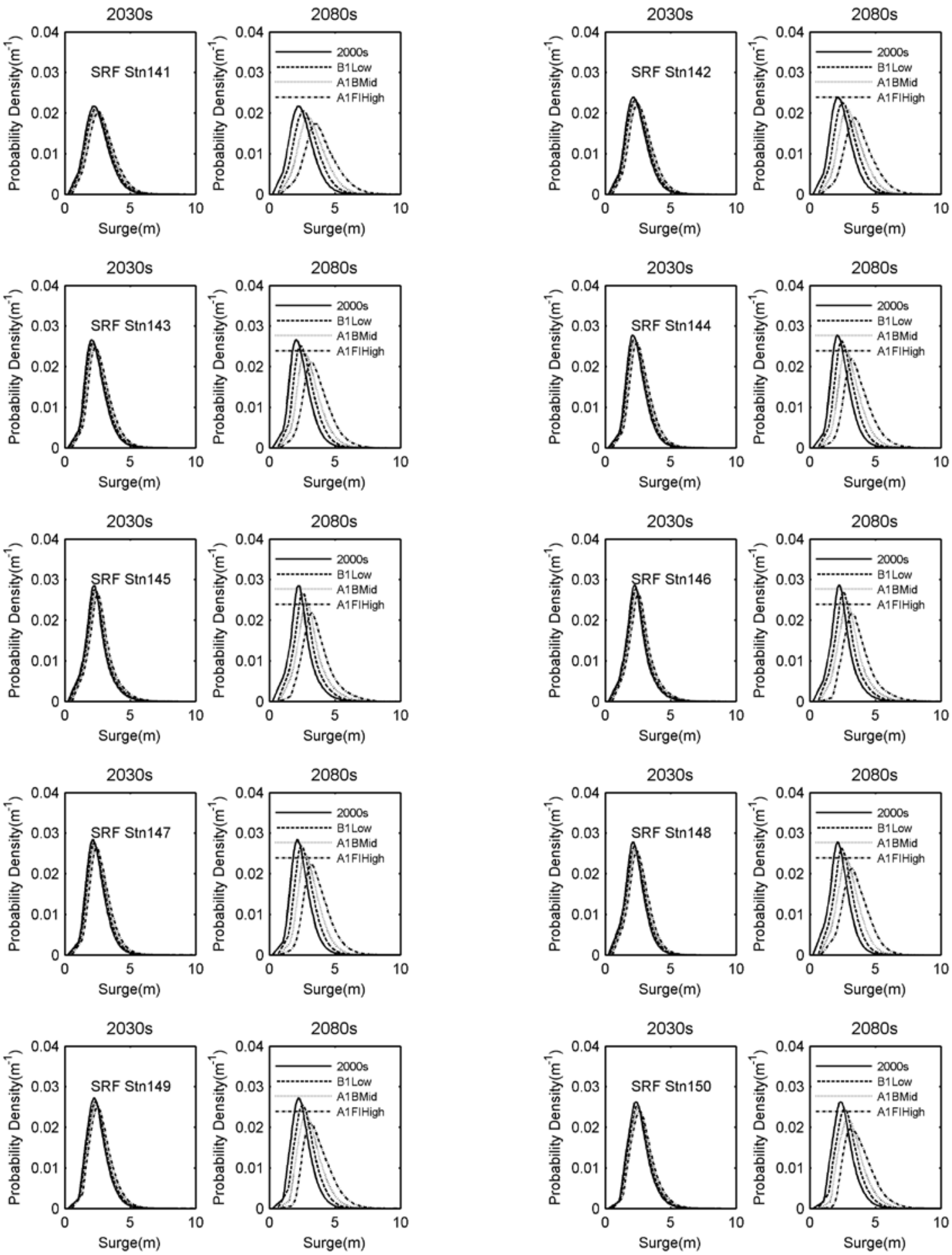

2030s
2080s
Probability Density(m^{-1})
Surge(m)
SRF Stn91
SRF Stn92
SRF Stn93
SRF Stn94
SRF Stn95
SRF Stn96
SRF Stn97
SRF Stn98
SRF Stn99
SRF Stn100
2000s
B1Low
A1BMid
A1FIHigh

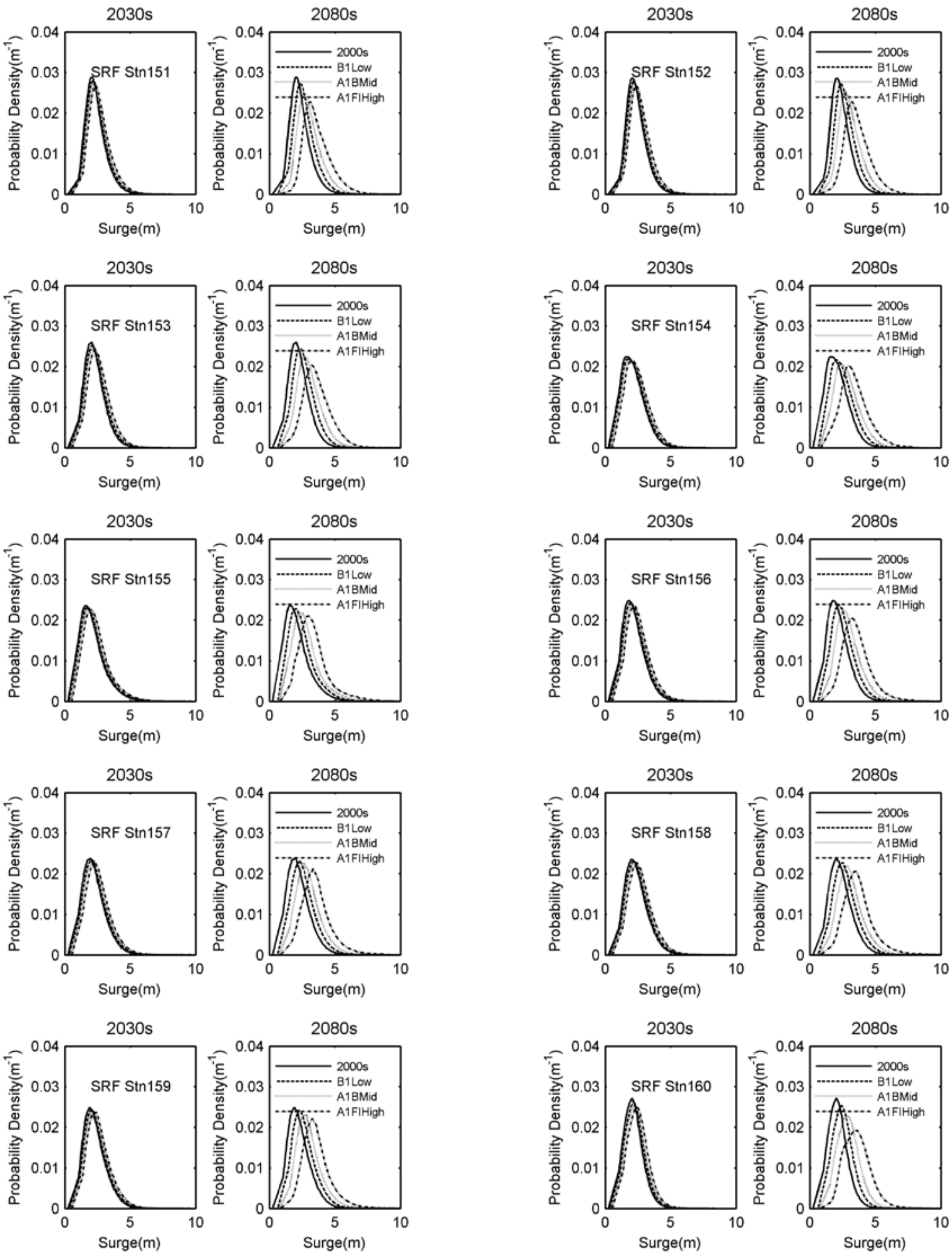

2030s
2080s
Probability Density(m^{-1})
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh
SRF Stn101
SRF Stn102
SRF Stn103
SRF Stn104
SRF Stn105
SRF Stn106
SRF Stn107
SRF Stn108
SRF Stn109
SRF Stn110


2030s
2080s
SRF Stn111
SRF Stn112
SRF Stn113
SRF Stn114
SRF Stn115
SRF Stn116
SRF Stn117
SRF Stn118
SRF Stn119
SRF Stn120
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)
0
0.01
0.02
0.03
0.04
0
5
10

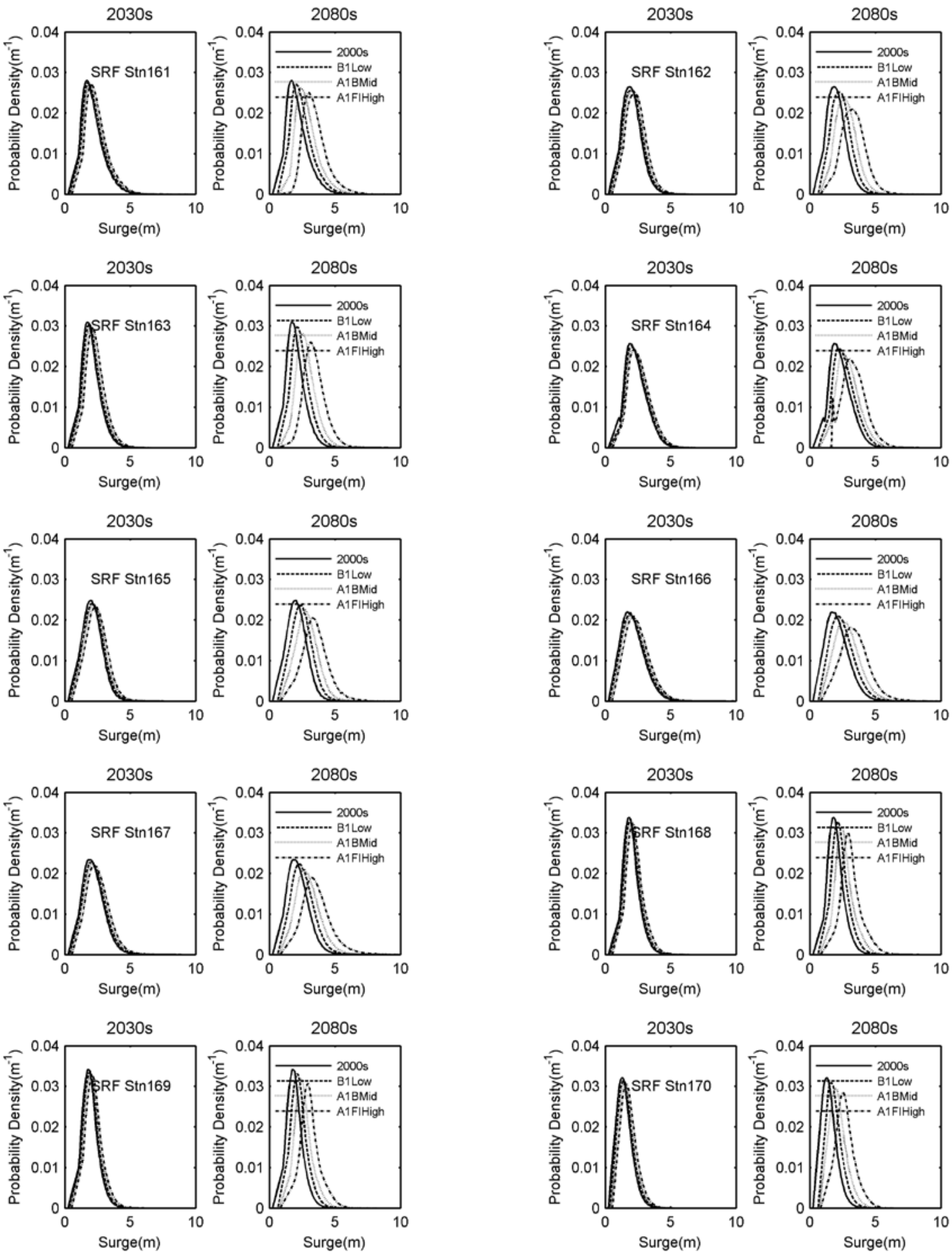

2030s
2080s
Probability Density(m-1)
Surge(m)
SRF Stn121
SRF Stn122
SRF Stn123
SRF Stn124
SRF Stn125
SRF Stn126
SRF Stn127
SRF Stn128
SRF Stn129
SRF Stn130
2000s
B1Low
A1BMid
A1FIHigh

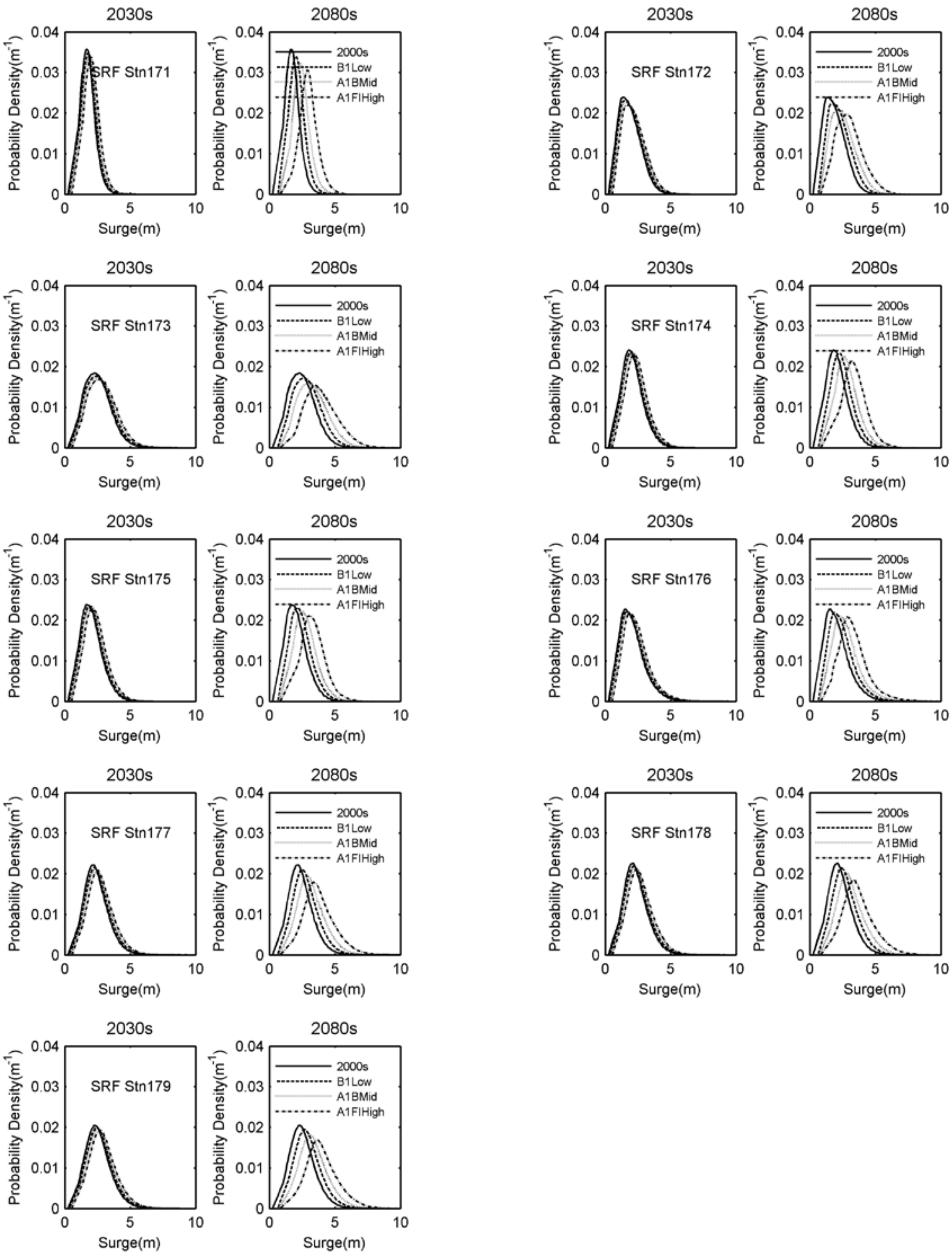

2030s
2080s
SRF Stn131
SRF Stn132
SRF Stn133
SRF Stn134
SRF Stn135
SRF Stn136
SRF Stn137
SRF Stn138
SRF Stn139
SRF Stn140
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)


2030s
2080s
Probability Density(m^{-1})
Surge(m)
2000s
B1Low
A1BMid
A1FIHigh
SRF Stn141
SRF Stn142
SRF Stn143
SRF Stn144
SRF Stn145
SRF Stn146
SRF Stn147
SRF Stn148
SRF Stn149
SRF Stn150



2030s
2080s
SRF Stn151
SRF Stn152
SRF Stn153
SRF Stn154
SRF Stn155
SRF Stn156
SRF Stn157
SRF Stn158
SRF Stn159
SRF Stn160
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)


2030s
2080s
SRF Stn161
SRF Stn162
SRF Stn163
SRF Stn164
SRF Stn165
SRF Stn166
SRF Stn167
SRF Stn168
SRF Stn169
SRF Stn170
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)


2030s
2080s
SRF Stn171
SRF Stn172
SRF Stn173
SRF Stn174
SRF Stn175
SRF Stn176
SRF Stn177
SRF Stn178
SRF Stn179
2000s
B1Low
A1BMid
A1FIHigh
Probability Density(m^{-1})
Surge(m)

APPENDIX G AFFECTED BUSINESS VS. RETURN PERIOD ANALYSIS

G.1. Corpus Christi, TX (Rank 1 Business: Food Services and Drinking Places)

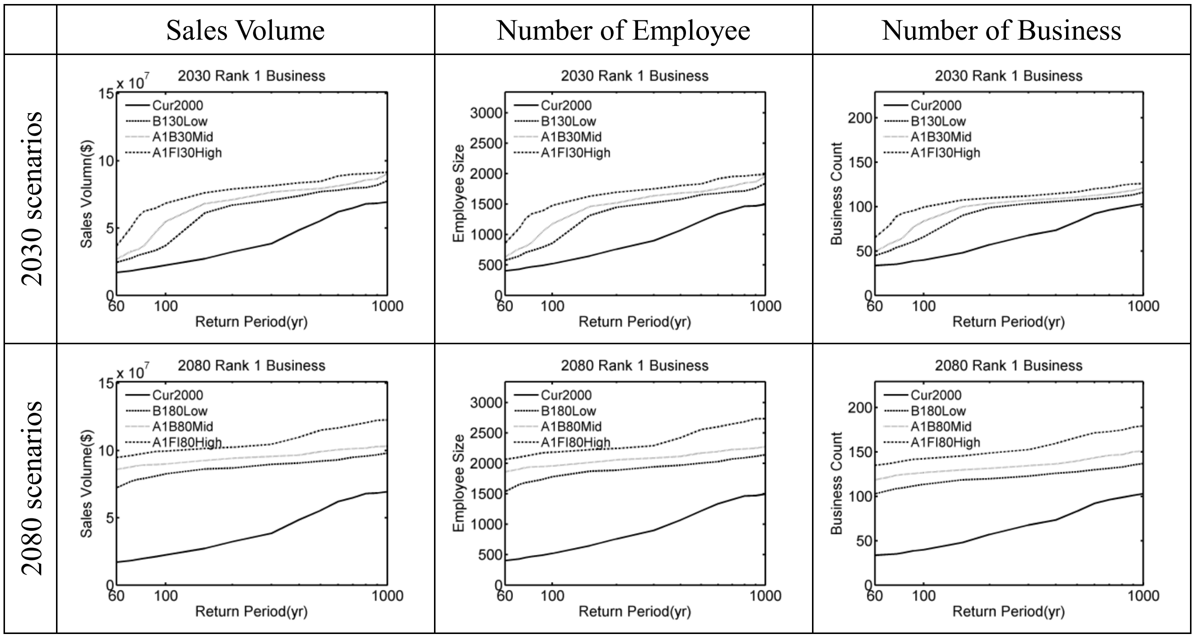


Figure G-1 Consider the Business below Future Sea Level as Complete Loss (Corpus Christi, Rank1 Business)

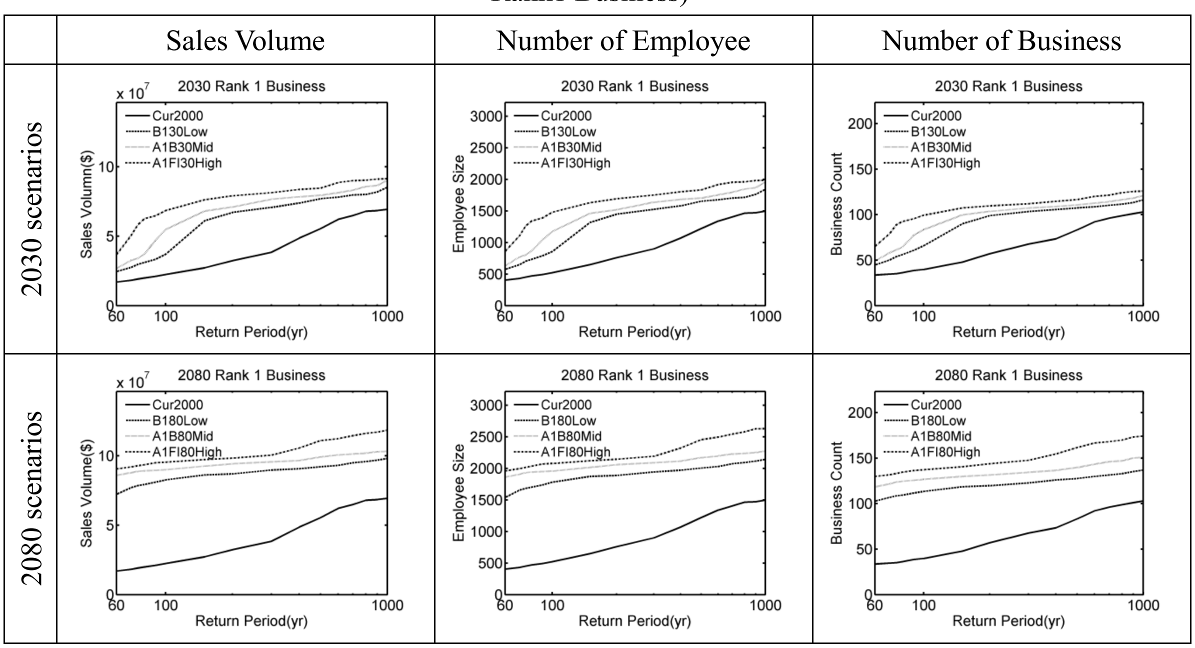


Figure G-2 Without Considering the Business below Future Sea Level (Corpus Christi, Rank1 Business)

G.2. Corpus Christi, TX (Rank 2 Business: Educational Services)

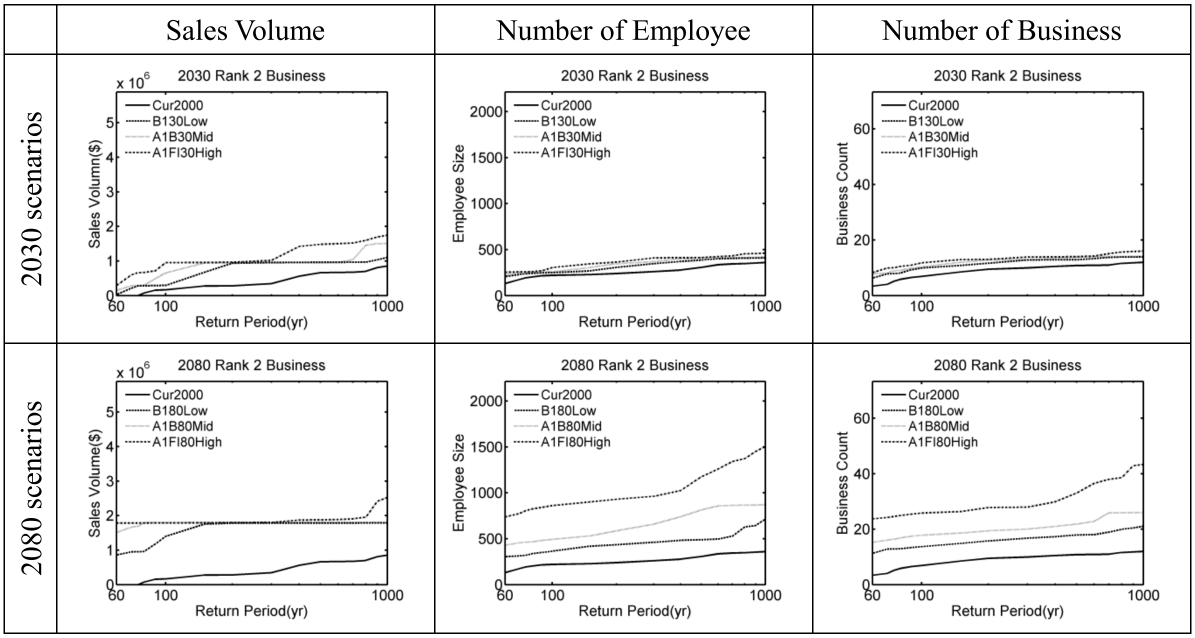


Figure G-3 Consider the Business below Future Sea Level as Complete Loss (Corpus Christi, Rank2 Business)

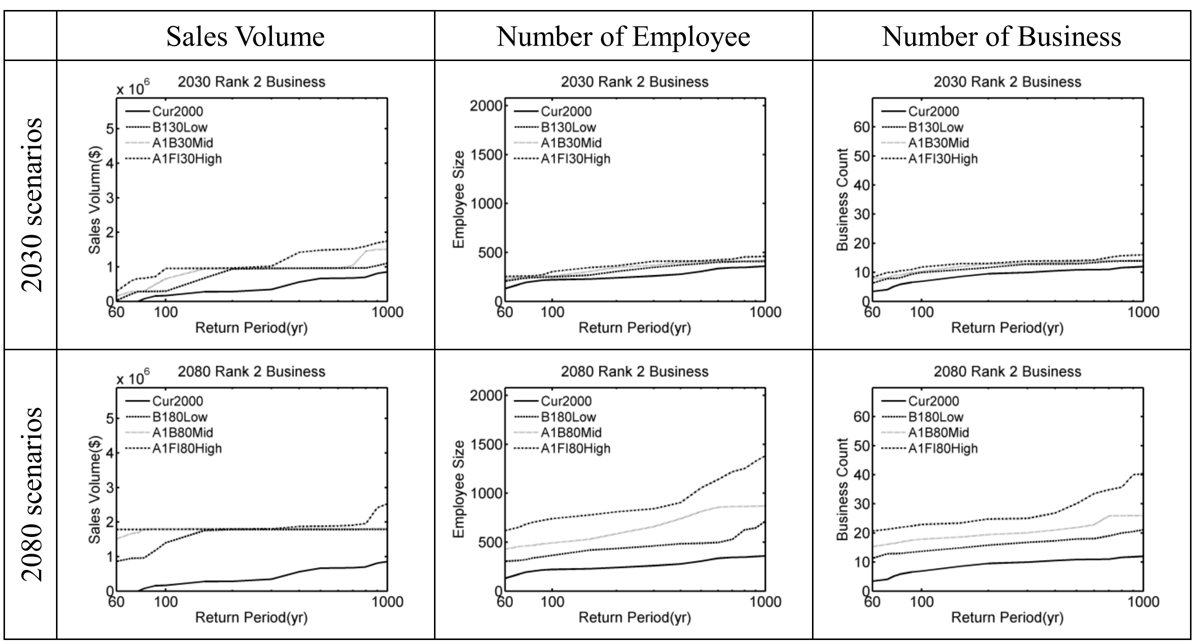


Figure G-4 Without Considering the Business below Future Sea Level (Corpus Christi, Rank2 Business)

G.3. Corpus Christi, TX (Rank 3 Business: Ambulatory Health Care Services)

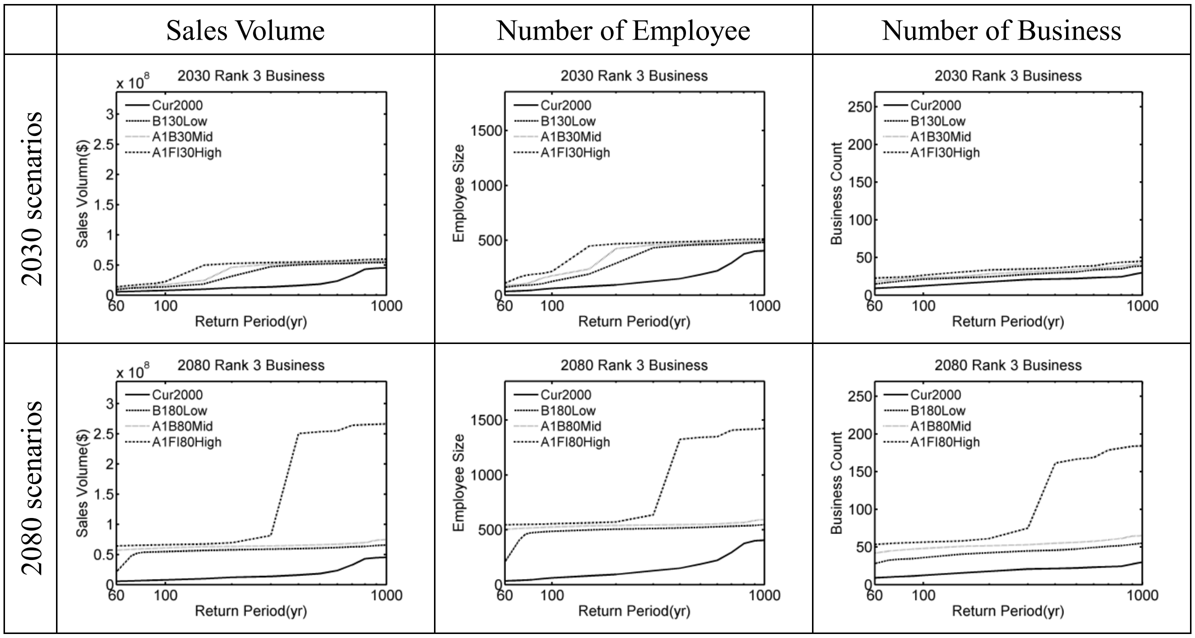


Figure G-5 Consider the Business below Future Sea Level as Complete Loss (Corpus Christi, Rank3 Business)

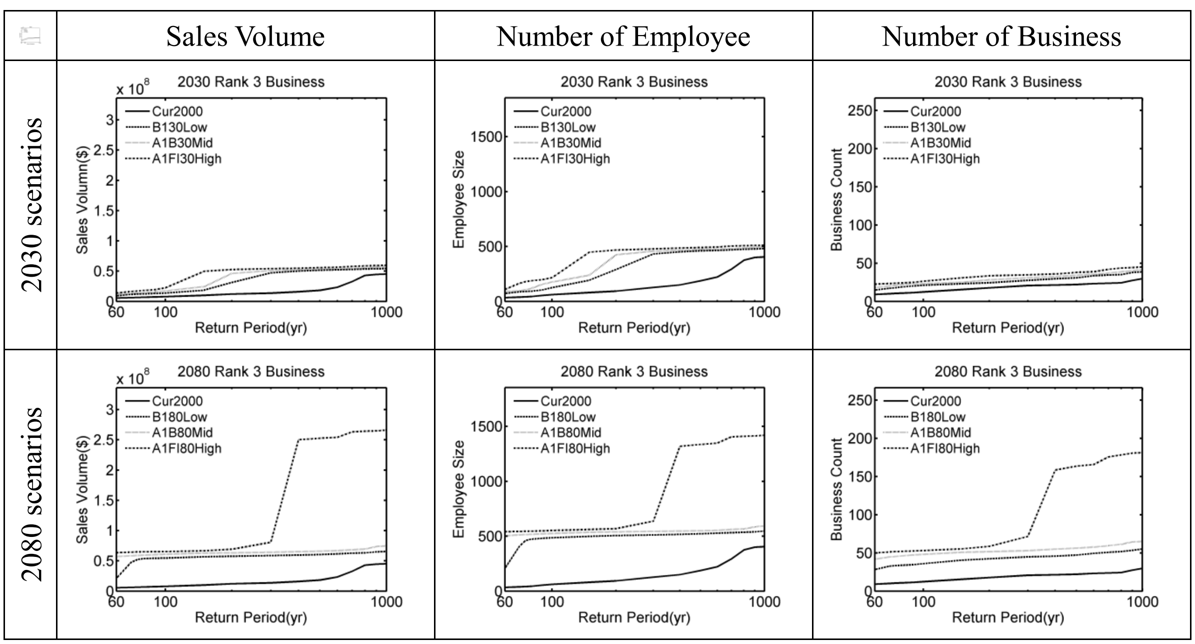


Figure G-6 Without Considering the Business Below Future Sea Level (Corpus Christi, Rank3 Business)

G.4. Corpus Christi, TX (Rank 4 Business: Specialty Trade Contractors)

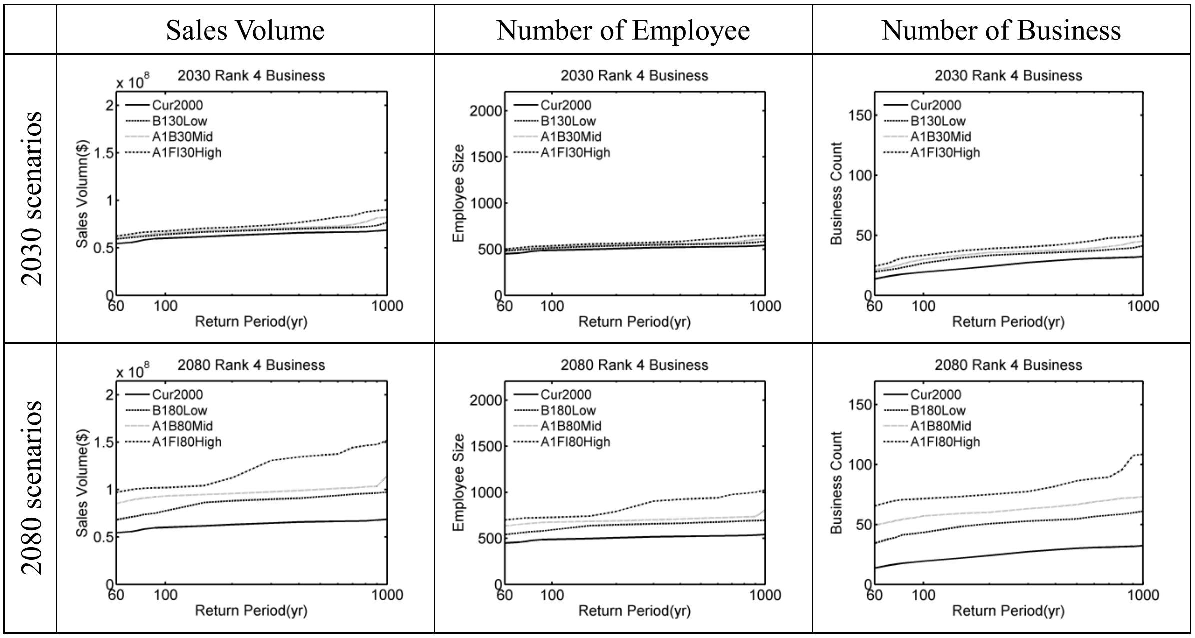


Figure G-7 Consider the Business below Future Sea Level as Complete Loss (Corpus Christi, Rank4 Business)

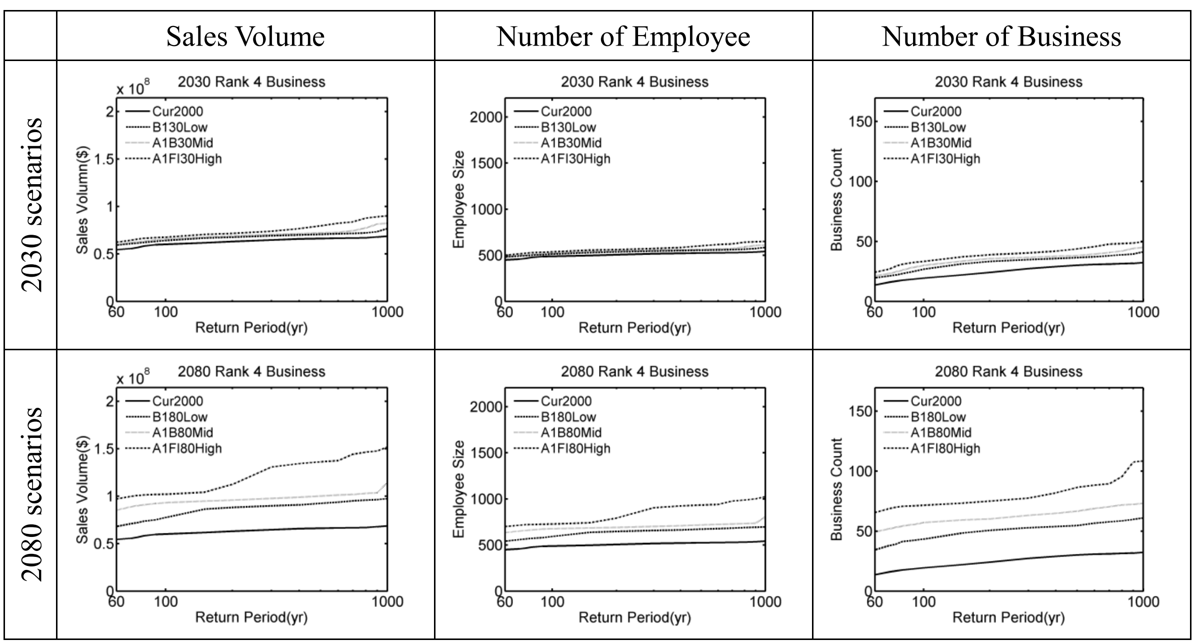


Figure G-8 Without Considering the Business below Future Sea Level (Corpus Christi, Rank4 Business)

G.5. Gulfport, MS (Rank 1 Business: Amusement, Gambling, and Recreation Industries)

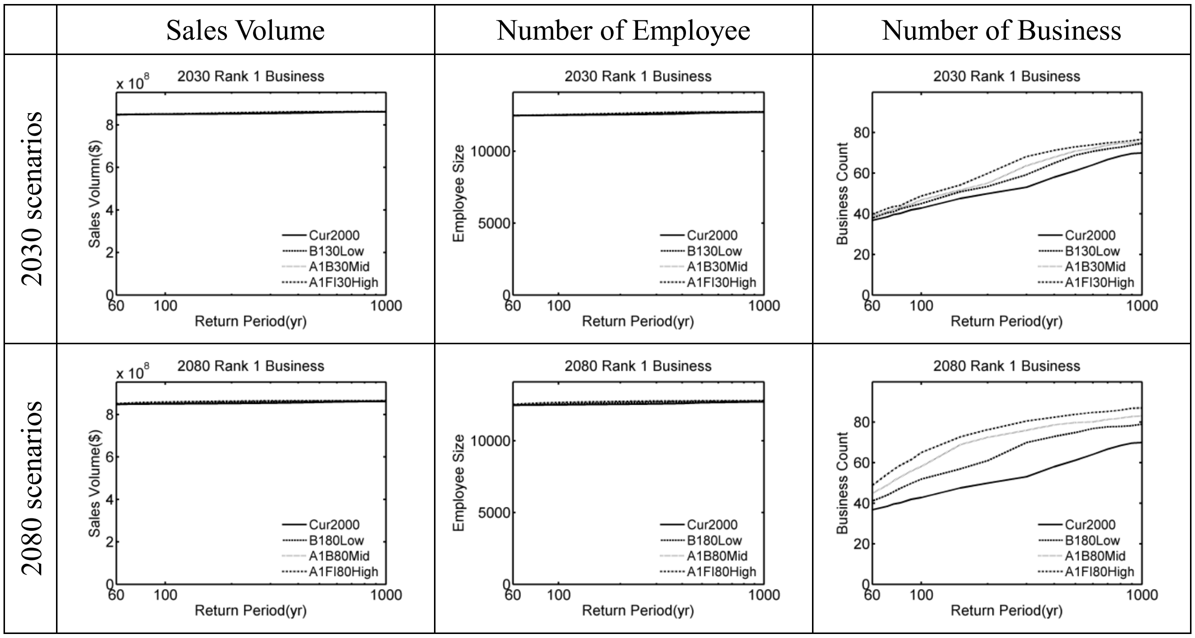


Figure G-9 Consider the Business below Future Sea Level as Complete Loss (Gulfport, Rank1 Business)

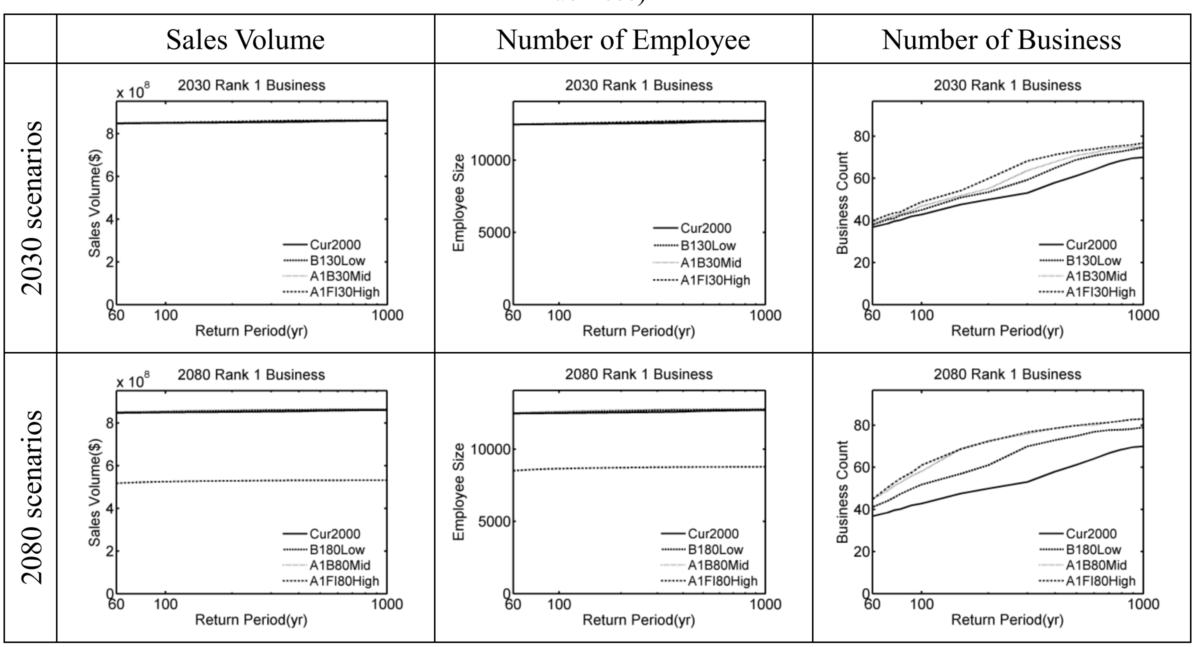


Figure G-10 Without Considering the Business below Future Sea Level (Gulfport, Rank1 Business)

G.6. Gulfport, MS (Rank 2 Business: Food Services and Drinking Places)

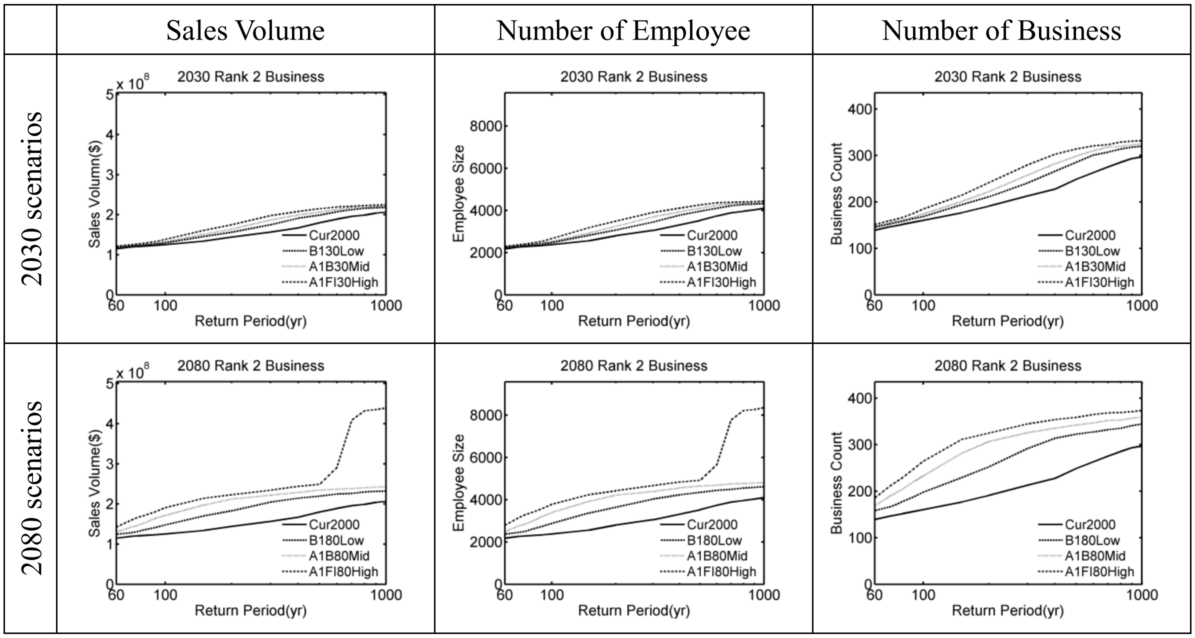


Figure G-11 Consider the Business below Future Sea Level as Complete Loss (Gulfport, Rank2 Business)

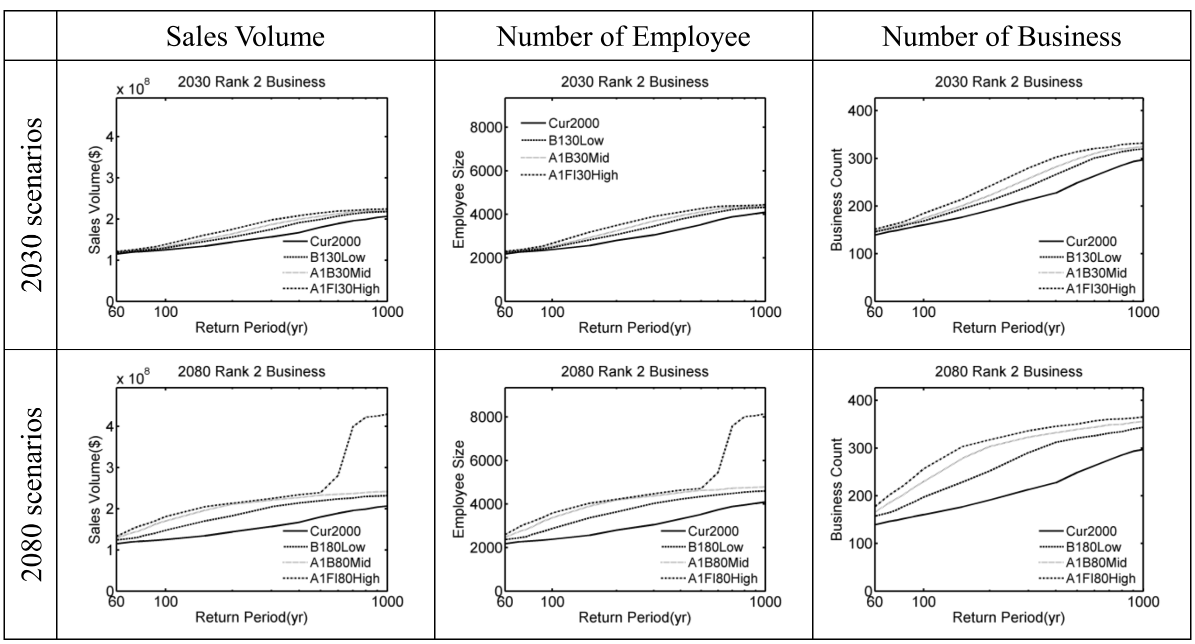


Figure G-12 Without Considering the Business below Future Sea Level (Gulfport, Rank2 Business)

G.7. Gulfport, MS (Rank 3 Business: Ambulatory Health Care Services)

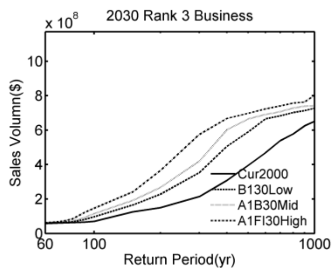
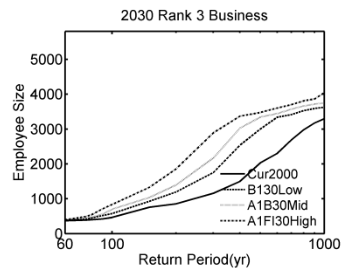
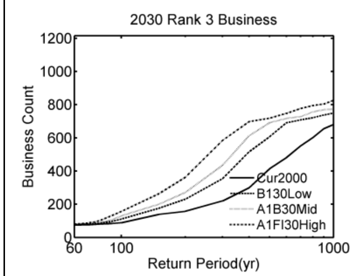
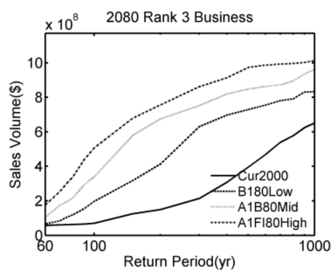
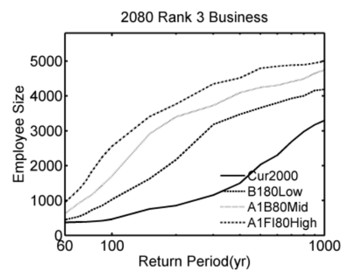
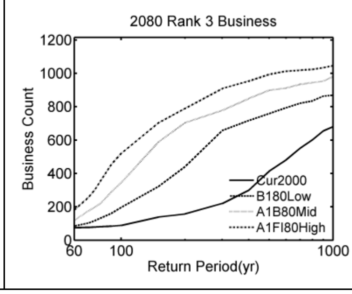
	Sales Volume	Number of Employee	Number of Business
2030 scenarios			
2080 scenarios			

Figure G-13 Consider the Business below Future Sea Level as Complete Loss (Gulfport, Rank3 Business)

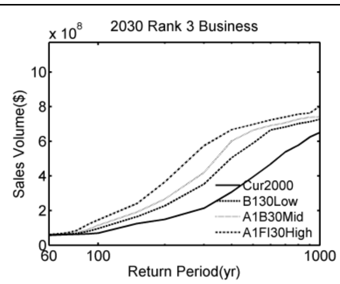
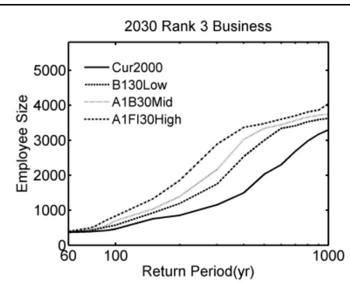
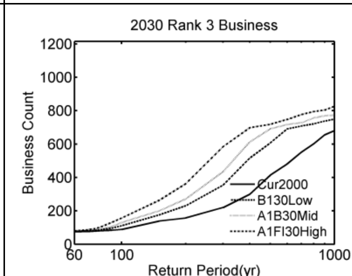
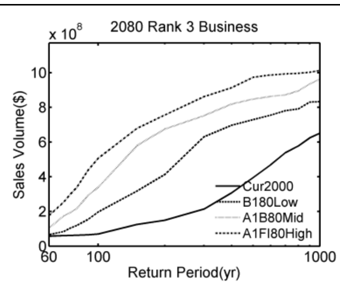
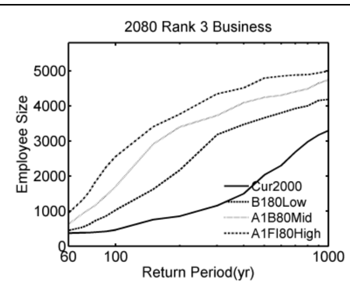
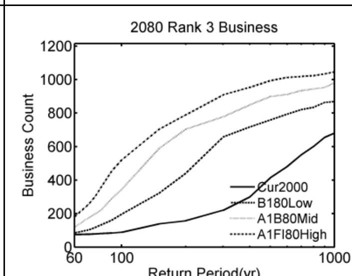
	Sales Volume	Number of Employee	Number of Business
2030 scenarios			
2080 scenarios			

Figure G-14 Without Considering the Business below Future Sea Level (Gulfport, Rank3 Business)

G.8. Gulfport, MS (Rank 4 Business: Educational Services)

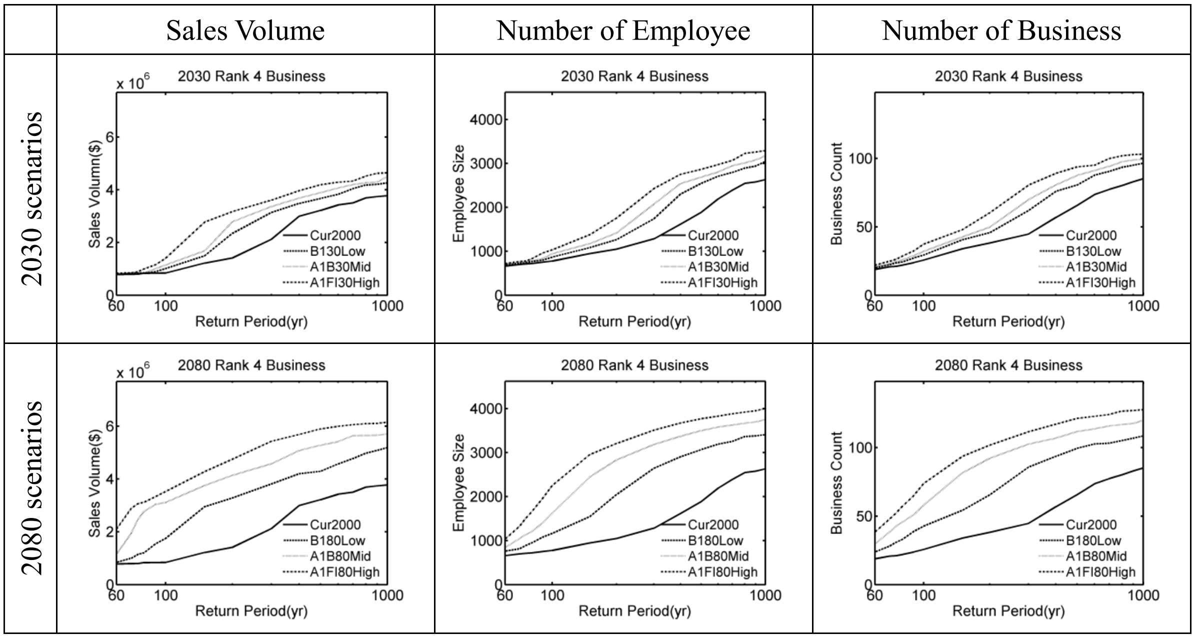


Figure G-15 Consider the Business below Future Sea Level as Complete Loss (Gulfport, Rank4 Business)

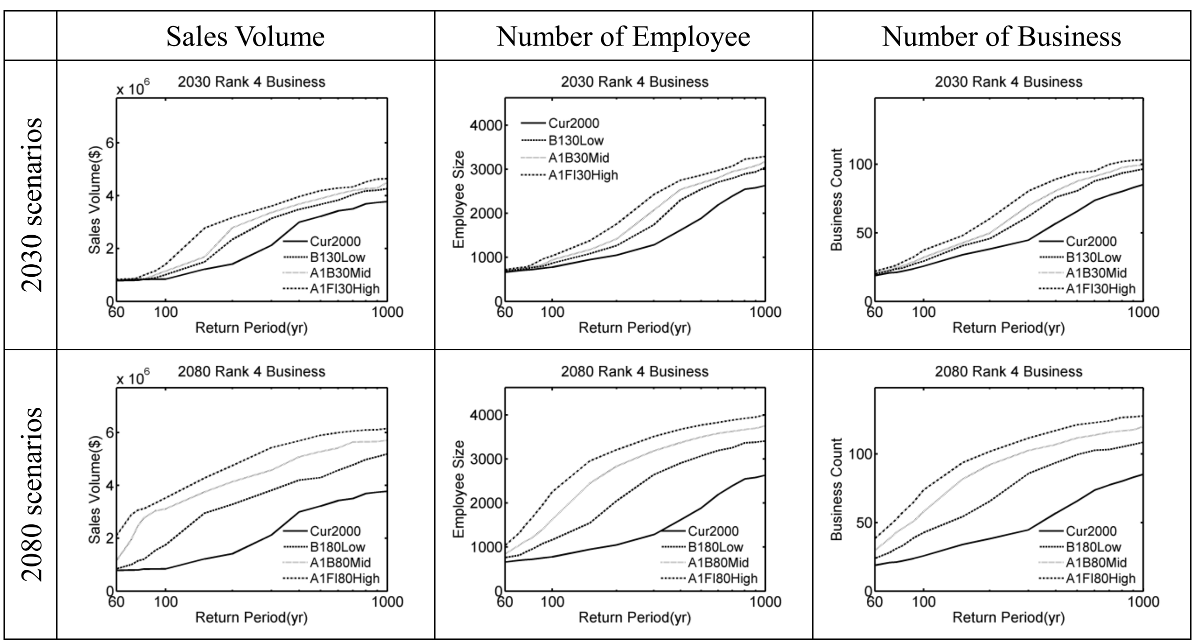


Figure G-16 Without Considering the Business Below Future Sea Level (Gulfport, Rank4 Business)

G.9. Panama, FL (Rank 1 Business: Food Services and Drinking Places)

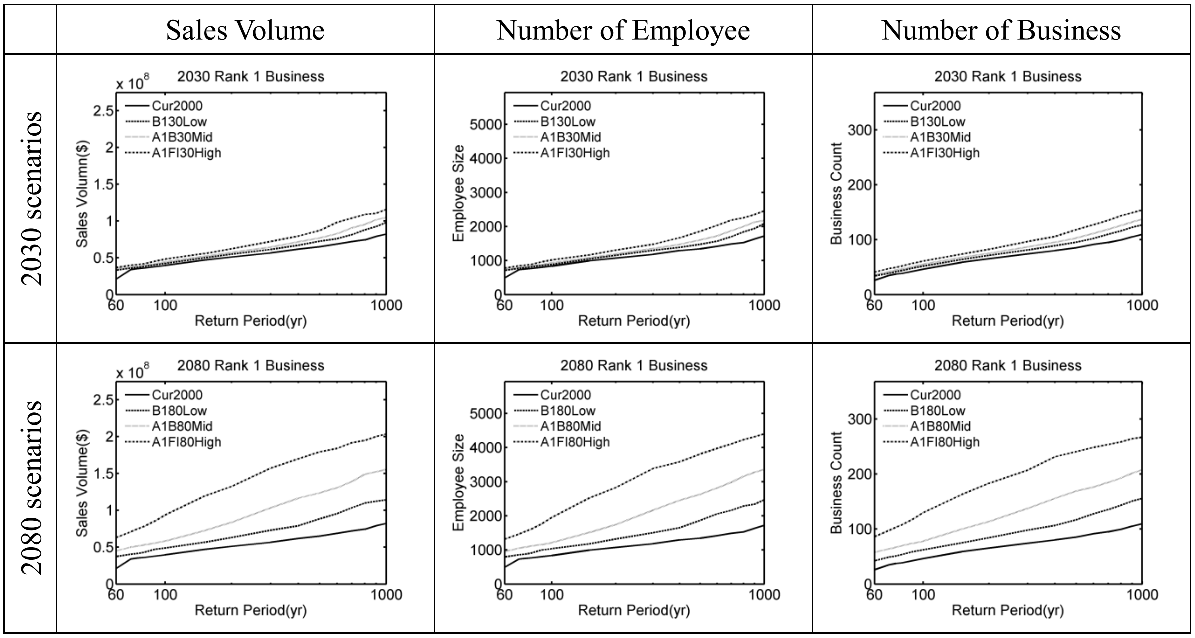


Figure G-17 Consider the Business below Future Sea Level as Complete Loss (Panama, Rank1 Business)

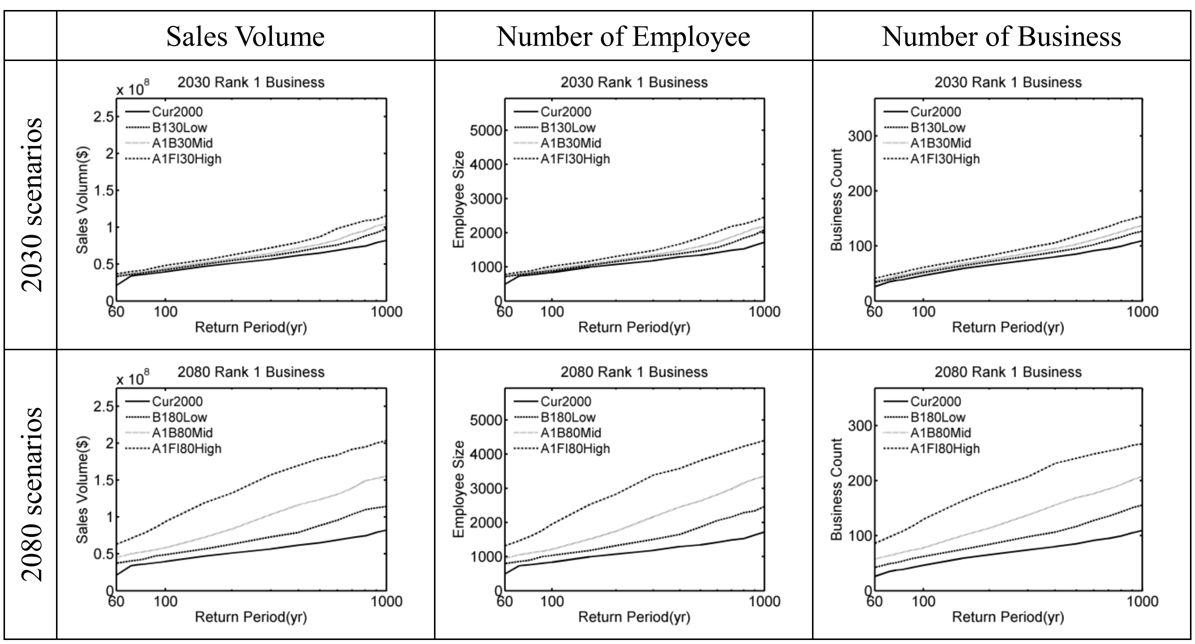


Figure G-18 Without Considering the Business below Future Sea Level (Panama, Rank1 Business)

G.10. Panama, FL (Rank 2 Business: Ambulatory Health Care Services)

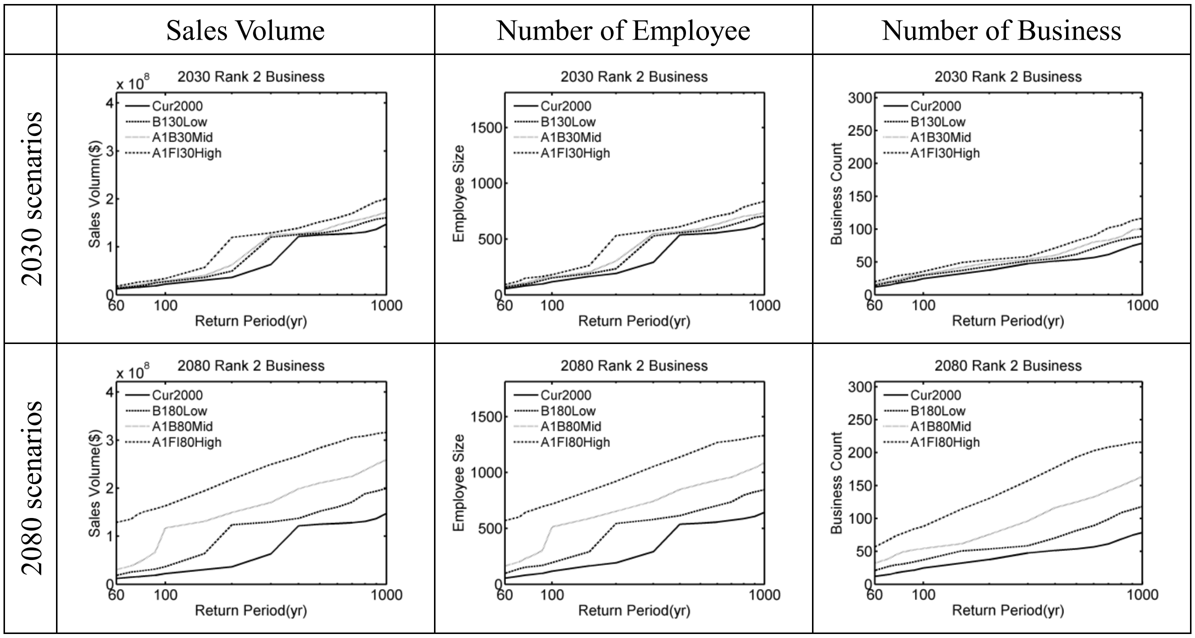


Figure G-19 Consider the Business below Future Sea Level as Complete Loss (Panama, Rank2 Business)

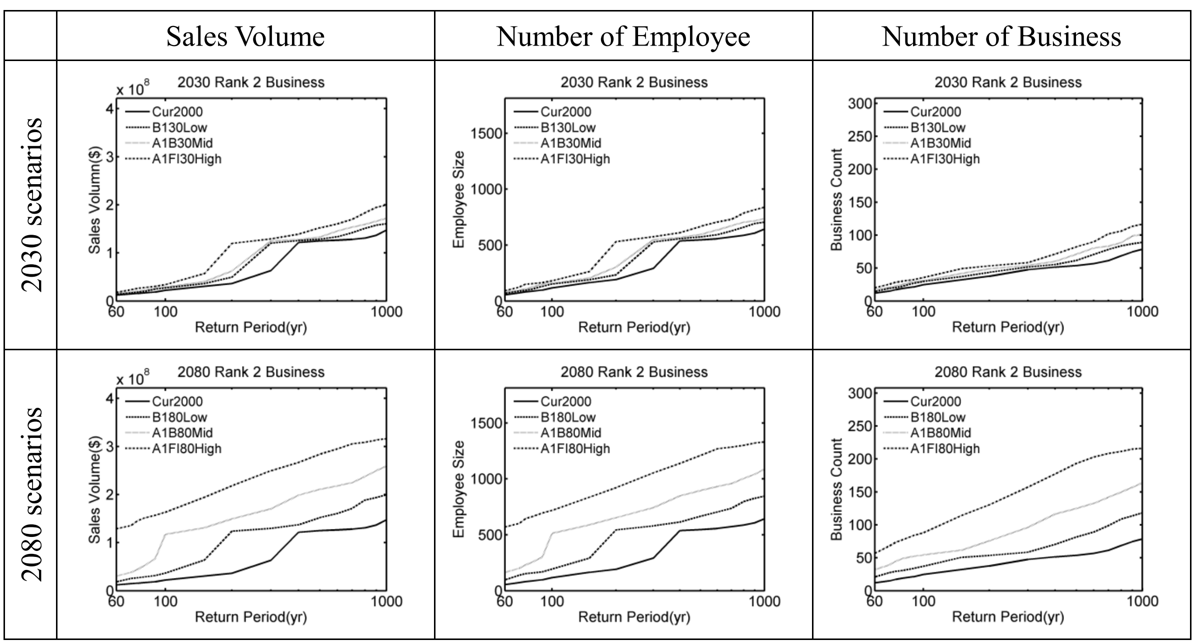


Figure G-20 Without Considering the Business below Future Sea Level (Panama, Rank2 Business)

G.11. Panama, FL (Rank 3 Business: Educational Services)

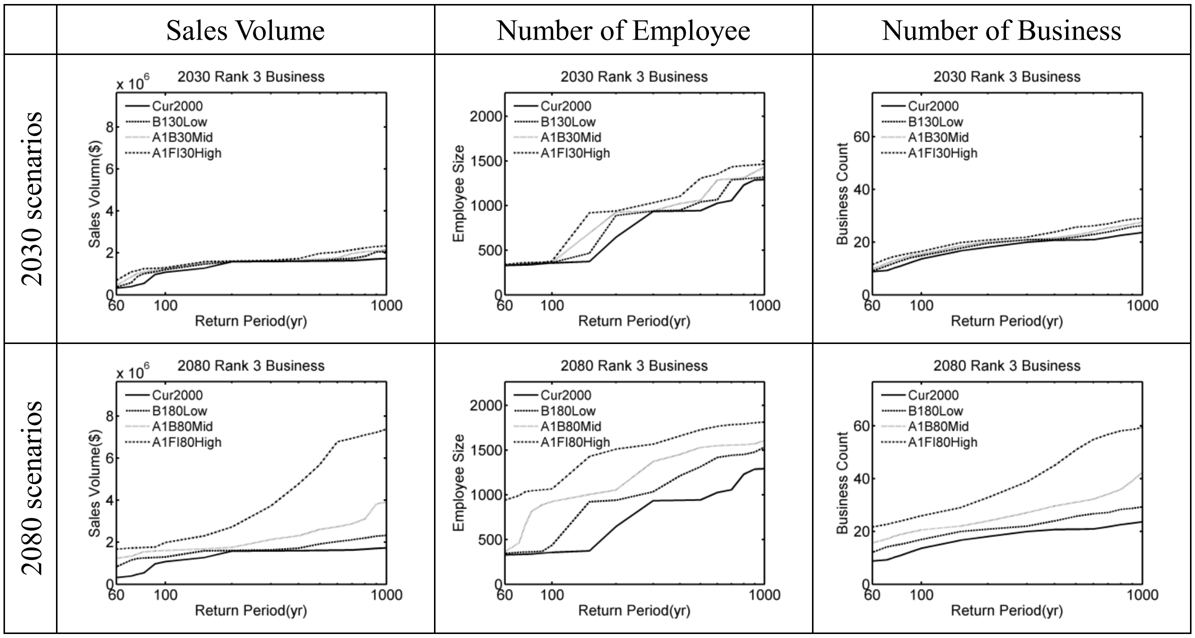


Figure G-21 Consider the Business below Future Sea Level as Complete Loss (Panama, Rank3 Business)

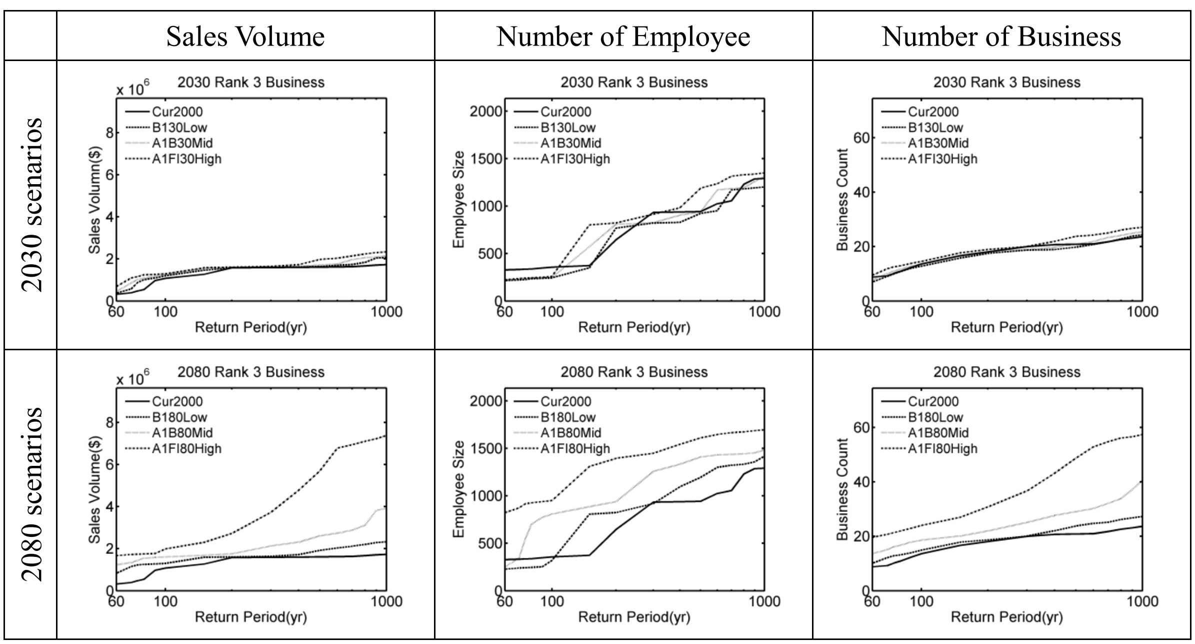


Figure G-22 Without Considering the Business below Future Sea Level (Panama, Rank3 Business)

G.12. Panama, FL (Rank 4 Business: Professional, Scientific, and Technical Services)

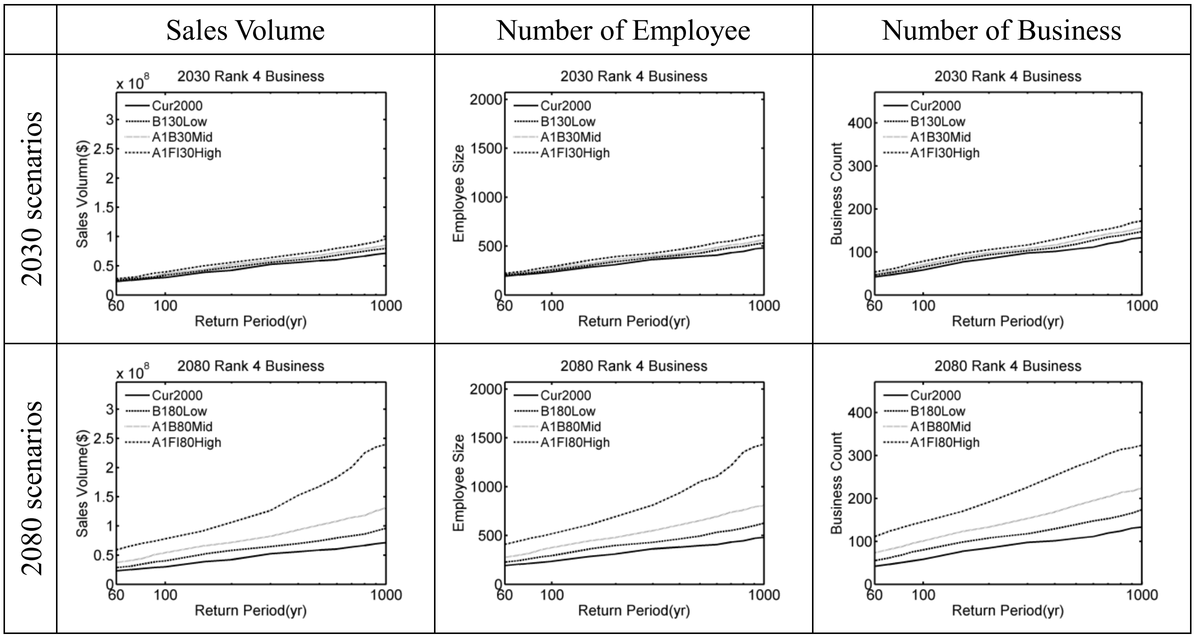


Figure G-23 Consider the Business below Future Sea Level as Complete Loss (Panama, Rank4 Business)

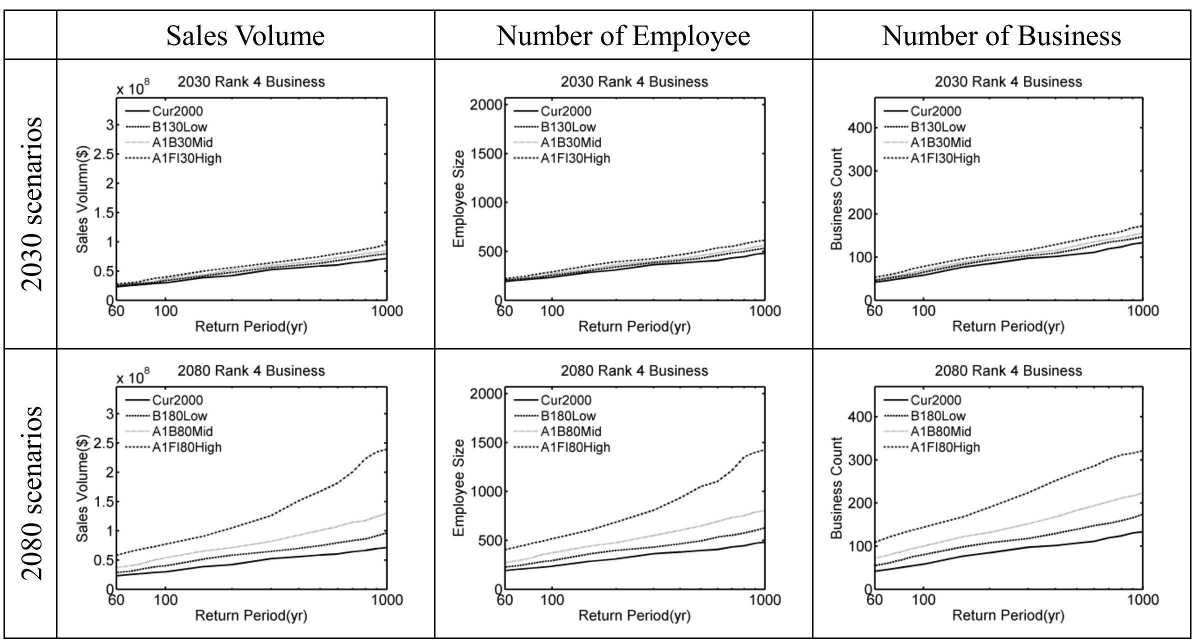


Figure G-24 Without Considering the Business below Future Sea Level (Panama, Rank4 Business)

APPENDIX H VISHURRICANE CODE

H.1. Default.aspx

```
//==============================================================================
//  Purpose
//      Into to SimHurricane
//
//  Functions
//      1. Create list of City Names                        ||  string[] CityNames = new
string[3]
//      2. Initialize current year variable                 ||  DateTime.Now.Year
//      3. Read in Hurricane Name table from database       ||  cn.ConnectionString
//
//  Sesssion Vars
//      1. "showMonitor"
//      2. "nowYear"
//      3. "CityNames"
//==============================================================================
//--Default
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
//--ADDED
using System.Web.UI.DataVisualization.Charting;
using System.Data;             //# For reading database
using System.Data.OleDb;       //# For reading database
//using System.Data.SqlClient;

public partial class Default2 : System.Web.UI.Page
{
   //**************************************************************************
   //--Set City Names
   static string[] CityNames = new string[3] { "CityA", "CityB", "CityC" };
//<====Specify City Names

   //#Check/Testing*************************************************************
   static DataTable HurNamT = new DataTable();

   Random Rnd = new Random();

   //====================================
   // PAGE EVENT
   //====================================
   protected void Page_PreInit(object sender, EventArgs e)
   {
```

```
        // SET showMonitor FLAG
        Session["showMonitor"] = 0;                 //★★★
        Session["nowYear"] = DateTime.Now.Year;     //★★★ # Retrievei current year
        Session["CurCO2"]=396.81;
        Session["GameName"] = "VisHurricane";
    }
    protected void Page_Init(object sender, EventArgs e)
    {
        // SHOW MONITORED PANEL
        Panel1.Visible = Convert.ToBoolean(Session["showMonitor"]);
        lblGameName.Text = Convert.ToString(Session["GameName"]);
        lblGameName2.Text = Convert.ToString(Session["GameName"]);
        lblGameName3.Text = Convert.ToString(Session["GameName"]);
        lblGameName4.Text = Convert.ToString(Session["GameName"]);
        lblGameName5.Text = Convert.ToString(Session["GameName"]);
    }

    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            //■■ Check in Monitor Panel
            lblshowMonitor.Text = Convert.ToString(Session["showMonitor"]);
            lblCurYr.Text = Convert.ToString(Session["nowYear"]);

            //----------------------------------------------------------
            // Access DBF and obtaion Hurricane Names
            //----------------------------------------------------------
            using (OleDbConnection cn = new OleDbConnection())
            {
                //cn.ConnectionString = "Provider=Microsoft.ACE.OLEDB.12.0;" +
                //                    "Data Source=" +
Server.MapPath("~/App_Data/HurricaneNames.accdb");
                cn.ConnectionString = "Provider=Microsoft.JET.OLEDB.4.0;" +
                                    "Data Source=" +
Server.MapPath("~/App_Data/HurricaneNames.mdb");
                cn.Open();
                DataSet ds = new DataSet();

                // DATAADAPTER
                OleDbDataAdapter daHurNam = new OleDbDataAdapter("SELECT * FROM
HurNam", cn);
                daHurNam.Fill(ds, "HurNam");

                // DATATABLE
                DataTable dtHurNam = new DataTable();
                dtHurNam = ds.Tables["HurNam"];

                // Store DataTable in Session
                Session["dtHurNam"] = dtHurNam;     //★★★
            }
```

```
            //■■ Check/Testing
            HurNamT=(DataTable)Session["dtHurNam"];     //# Cvt Session to DataTable
            lblHurNam.Text =
HurNamT.Rows[Rnd.Next(0,20)][Rnd.Next(2,7)].ToString();
        }
    }

    //=====================================
    // GO TO NEXT PAGE
    //=====================================
    protected void btnNext_Click(object sender, EventArgs e)
    {
        //--Passing Vars
        Session["CityNames"]=CityNames;
        //--Go to next page
        Response.Redirect("~/Page02.aspx");
    }
}
```

H.2. Page02.aspx

```
//==================================================================
//  Purpose
//      Show game steps
//  Functions
//      1. Passing showMonitor var
//      2. Redirect to next page
//==================================================================
//--Default
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Default2 : System.Web.UI.Page
{
    //====================================
    // PAGE EVENT
    //====================================
    protected void Page_Init(object sender, EventArgs e)
    {
        Panel1.Visible = Convert.ToBoolean(Session["showMonitor"]);
        lblGameName.Text = Convert.ToString(Session["GameName"]);
    }

    protected void Page_Load(object sender, EventArgs e)
    {
        //■■■ Check in Monitor Panel
        //lblCurYr.Text = Convert.ToString(Session["nowYear"]);   //#
Session("nowYear) defined at Default page

    }

    //====================================
    // GO TO NEXT PAGE
    //====================================
    protected void btnNext_Click(object sender, EventArgs e)
    {
        Response.Redirect("~/Page03.aspx");
    }
}
```

H.3. Page03.aspx

```
//=============================================================================
//  Purpose
//      1. Display city and coast image                            ||  image.ImageUrl
//      2. Display city description with corresponding city image   ||  MultiView  +
RadioButtonList
//  Function
//=============================================================================
//--Default
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Default2 : System.Web.UI.Page
{
    //======================================
    // PAGE EVENTS
    //======================================
    protected void Page_Init(object sender, EventArgs e)
    {
        Panel1.Visible = Convert.ToBoolean(Session["showMonitor"]); //★★ Defined
in Default
        lblGameName.Text = Convert.ToString(Session["GameName"]);   //★★ Defined
in Default
    }

    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack) //# First time load this page
        {
            //■■■ Check in Monitor Panel
            //lblCurYr.Text = Convert.ToString(Session["nowYear"]);   //#
Session("nowYear) defined at Default page

            //--Default setting
            Image2.ImageUrl="~/Image/City/City1.png";
            MultiView1.SetActiveView(View1);
        }

    }
    //======================================
    // WEB CONTROL EVENTS
    //======================================
    protected void RadioButtonList1_SelectedIndexChanged(object sender, EventArgs
e)
    {
```

```
        switch (RadioButtonList1.SelectedIndex) {
            case 0:
                Image2.ImageUrl = "~/Image/City/City1.png";
                Image1.ImageUrl = "~/Image/City/CityA2.png";
                MultiView1.SetActiveView(View1);
                break;
            case 1:
                Image2.ImageUrl = "~/Image/City/City2.png";
                Image1.ImageUrl = "~/Image/City/CityB2.png";
                MultiView1.SetActiveView(View2);
                break;
            case 2:
                Image2.ImageUrl = "~/Image/City/City3.png";
                Image1.ImageUrl = "~/Image/City/CityC2.png";
                MultiView1.SetActiveView(View3);
                break;
            default:
                break;
        }
    }
    //=====================================
    // GO TO NEXT PAGE
    //=====================================
    protected void btnNext_Click(object sender, EventArgs e)
    {
        Response.Redirect("~/Page04.aspx");
    }
}
```

H.4. Page04.aspx

```
//==============================================================================
// Purpose
//      1. Display IPCC Scenarios (images)            ||  image.ImageUrl
//      2. Display IPCC Scenarios description          ||  MultiView  +
RadioButtonList
// Function
//==============================================================================
//--Default
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Default2 : System.Web.UI.Page
{
    //=====================================
    // PAGE EVENTS
    //=====================================
    protected void Page_Init(object sender, EventArgs e)
    {
        Panel1.Visible = Convert.ToBoolean(Session["showMonitor"]);    //★★
Define in Default
    }
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            //--Initial image and text
            ImgTmp.ImageUrl = "./Image/temp/fig1.png";  //* .ImageUrl
            ImgCO2.ImageUrl = "./Image/CO2/fig1.png";
            MultiView1.SetActiveView(View1);          //* MultiView
        }
    }

    //=====================================
    // WEB CONTROL EVENTS
    //=====================================

    protected void RadioButtonList1_SelectedIndexChanged(object sender, EventArgs
e)
    {
        switch (RadioButtonList1.SelectedIndex)
        {
            case 0:
                MultiView1.SetActiveView(View1);
                // DISPLAY THE TMP FIG
```

```
                ImgTmp.ImageUrl = "./Image/temp/fig1.png";
                ImgCO2.ImageUrl = "./Image/CO2/fig1.png";
                break;
            case 1:
                MultiView1.SetActiveView(View2);
                // DISPLAY THE TMP FIG
                ImgTmp.ImageUrl = "./Image/temp/fig2.png";
                ImgCO2.ImageUrl = "./Image/CO2/fig2.png";
                break;
            case 2:
                MultiView1.SetActiveView(View3);
                // DISPLAY THE TMP FIG
                ImgTmp.ImageUrl = "./Image/temp/fig3.png";
                ImgCO2.ImageUrl = "./Image/CO2/fig3.png";
                break;
            case 3:
                MultiView1.SetActiveView(View4);
                // DISPLAY THE TMP FIG
                ImgTmp.ImageUrl = "./Image/temp/fig4.png";
                ImgCO2.ImageUrl = "./Image/CO2/fig4.png";
                break;
            case 4:
                MultiView1.SetActiveView(View5);
                // DISPLAY THE TMP FIG
                ImgTmp.ImageUrl = "./Image/temp/fig5.png";
                ImgCO2.ImageUrl = "./Image/CO2/fig5.png";
                break;
            case 5:
                MultiView1.SetActiveView(View6);
                // DISPLAY THE TMP FIG
                ImgTmp.ImageUrl = "./Image/temp/fig6.png";
                ImgCO2.ImageUrl = "./Image/CO2/fig6.png";
                break;
            default:
                break;
        }
    }

    //=====================================
    // GO TO NEXT PAGE
    //=====================================
    protected void btnNext_Click(object sender, EventArgs e)
    {
        Response.Redirect("~/Page05.aspx");
    }
}
```

H.5. Page05.aspx

```
//=============================================================================
//  Purpose
//     1. Let the user to choose SimPeriod, Scenario and allocate resources (select
and store)
//
//  Function
//     1. Store OptRlt of web control as Session Variable
//     2. Before leaving this page, retrieve City HurDmg Tb and IPCC sno Table (CO2
and Tmp)
//
//=============================================================================
//--Default
using System;
using System.Collections.Generic;
using System.Collections;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
//--ADDED
using System.Collections;
using System.Drawing;
using System.Web.UI.DataVisualization.Charting;
using System.Data;
using System.Data.OleDb;

public partial class test02 : System.Web.UI.Page
{
    //=======================================================
    // DECLARE PUBLIC VARIABLES
    //=======================================================

//#############################################################################
###
    //static double CpT = new double();

    static int simIPCC = new int();        //# The IPCC scenario choosed
    static int[] MitArr = new int[3];      //# Store the value of mitigation
    static int[] PopArr = new int[3];      //# Store the value of population


//#############################################################################
###
    static ArrayList DmgSrsA = new ArrayList(); // Series Dmg
    static ArrayList DmgSrsB = new ArrayList(); // Series Dmg
    static ArrayList DmgSrsC = new ArrayList(); // Series Dmg
    static ArrayList DmgSrsAr = new ArrayList(); // Series Dmg reduced
    static ArrayList DmgSrsBr = new ArrayList(); // Series Dmg reduced
```

```
    static ArrayList DmgSrsCr = new ArrayList(); // Series Dmg reduced

    static ArrayList PopSrsA = new ArrayList(); // Series Pop
    static ArrayList PopSrsB = new ArrayList(); // Series Pop
    static ArrayList PopSrsC = new ArrayList(); // Series Pop
    static ArrayList PopSrsAr = new ArrayList(); // Series Pop reduced
    static ArrayList PopSrsBr = new ArrayList(); // Series Pop reduced
    static ArrayList PopSrsCr = new ArrayList(); // Series Pop reduced

    static ArrayList AraSrsA = new ArrayList(); // Series Ara
    static ArrayList AraSrsB = new ArrayList(); // Series Ara
    static ArrayList AraSrsC = new ArrayList(); // Series Ara

    static ArrayList CpTSrs = new ArrayList(); // Series Srs Theoretical Cp
    static ArrayList CpRSrs = new ArrayList(); // Series Srs True Cp


//###############################################################################
###
    static double AnnBgt;      // Global variable Annual Budget


//###############################################################################
###
    //=======================================================
    // PAGE EVENTS
    //=======================================================
    //************************************************************
    // Page_Init:
    //      1. Initialize Session["simYear"]
    //      2. Default setting for Chart1 (the mitigation chart)
    //*************************************************************
    protected void Page_Init(object sender, EventArgs e)   //# THIS WON'T AFFECT LOAD
EVENT...
    {
        // TURN ON/OFF MONITOR PANEL
        Panel1.Visible = Convert.ToBoolean(Session["showMonitor"]);
        //Panel1.Visible = Convert.ToBoolean(1);

        // Session Vars
        Session["simYear"] = DateTime.Now.Year;              //# Ini simYear using
current year   //★★★
    }
    //*************************************************************
    // Page_Load:
    //      1. Default display of Image1 (Sno Tmp)
    //      2. Clear contents of vars
    //*************************************************************
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
```

```
        {
            //-----------------------------------------------------
            // ## Initialize Resource/ Annual Budget
            //-----------------------------------------------------
            AnnBgt = getBudget(); Session["AnnBgt"]=AnnBgt;  //# Get annual budget
and pass by session
            lblAnnBgt.Text = Convert.ToString(AnnBgt);
            lblResA.Text = Convert.ToString(0);
            lblResB.Text = Convert.ToString(0);
            lblResC.Text = Convert.ToString(0);
            //DropDownList1.SelectedValue = Convert.ToString(100);

            //-------------------------------------------------------
            //%% DEFAULT CHART PROPERTIES
            //-------------------------------------------------------
            Chart1.ChartAreas[0].AxisX.MajorGrid.Enabled = false;
            Chart1.ChartAreas[0].AxisY.MajorGrid.Enabled = false;
            //% Set ylabel
            Chart1.ChartAreas[0].AxisY.Title = "% of total budget";
            Chart1.ChartAreas[0].AxisY.TitleFont = new Font("default", 12);
            //% SET CHART LIMIT
            Chart1.ChartAreas[0].AxisX.Maximum = 4;
            Chart1.ChartAreas[0].AxisX.Minimum = -1;
            Chart1.ChartAreas[0].AxisY.Maximum = 100;
            Chart1.ChartAreas[0].AxisY.Minimum = 0;
            //% CREATE DATA POINTS (OTHERWISE, SET UP IN PROPERTY WINDOW)
            Chart1.Series[0].Points.Add(new DataPoint(0, 0) { AxisLabel = "CityA" });
//# THE THIRD POINT HAS TO BE ADDED
            Chart1.Series[0].Points.Add(new DataPoint(1, 0) { AxisLabel = "CityB" });
//# THE THIRD POINT HAS TO BE ADDED
            Chart1.Series[0].Points.Add(new DataPoint(2, 0) { AxisLabel = "CityC" });
//# THE THIRD POINT HAS TO BE ADDED
            Chart1.Series[0].Points.Add(new DataPoint(3, AnnBgt) { AxisLabel =
"Quota" });   //# THE THIRD POINT HAS TO BE ADDED
            //--Initialize CpT
            //CpT=940;

            //-----------------------------------
            // 1. Default Display of temperature
            //-----------------------------------
            MultiView1.SetActiveView(View1);
            Image1.ImageUrl = "~/Image/temp/fig1.png";

            //------------------------------
            // 2. Clear Array Vars
            //------------------------------
            Array.Clear(MitArr, 0, MitArr.Length);
            Array.Clear(PopArr, 0, PopArr.Length);

            //------------------------------
            // 3. Clear ArrlyList Vars
```

```
            //----------------------------
            DmgSrsA.Clear();
            DmgSrsB.Clear();
            DmgSrsC.Clear();
            DmgSrsAr.Clear();
            DmgSrsBr.Clear();
            DmgSrsCr.Clear();

            PopSrsA.Clear();
            PopSrsB.Clear();
            PopSrsC.Clear();
            PopSrsAr.Clear();
            PopSrsBr.Clear();
            PopSrsCr.Clear();

            AraSrsA.Clear();
            AraSrsB.Clear();
            AraSrsC.Clear();

            //■■■Check
            lblDmgSrsA.Text = DmgSrsA.Count.ToString();

            //--ArrayList to Store CpT, CpR
            CpTSrs.Clear();
            CpRSrs.Clear();
            //--------------------------------------------------
            // 4. Initialize Sessions
            //--------------------------------------------------
            Session["simPrd"] = RadioButtonList1.SelectedValue;    //# SPECIFY SIMULATION PERIOD  //★★★ RadioButton
            Session["simIPCC"] = RadioButtonList2.SelectedValue;    //# SPECIFY IPCC SCENARIO //★★★ RadioButton


        }
    }

    //========================================================
    // WEB CONTROL EVENTS
    //========================================================
    //********************************************************
    // RadioButtonList1:
    //     1. Select and change the View (No data storage)
    //********************************************************
    protected void RadioButtonList1_SelectedIndexChanged(object sender, EventArgs e)
    {

        switch (RadioButtonList1.SelectedIndex)
        {
            case 0:
```

```
                MultiView1.SetActiveView(View1);
                break;
            case 1:
                MultiView1.SetActiveView(View2);
                break;
            case 2:
                MultiView1.SetActiveView(View3);
                break;
            default:
                break;
        }
        Session["simPrd"] = RadioButtonList1.SelectedValue;     //# SPECIFY
SIMULATION PERIOD  //★★★
    }

    //*******************************************************
    // RadioButtonList2:
    //      1. Select and change the figures (No data storage)
    //*******************************************************
    protected void RadioButtonList2_SelectedIndexChanged(object sender, EventArgs
e)
    {
        switch (RadioButtonList2.SelectedIndex)
        {
            case 0:
                Image1.ImageUrl = "~/Image/temp/fig1.png";
                break;
            case 1:
                Image1.ImageUrl = "~/Image/temp/fig2.png";
                break;
            case 2:
                Image1.ImageUrl = "~/Image/temp/fig3.png";
                break;
            case 3:
                Image1.ImageUrl = "~/Image/temp/fig4.png";
                break;
            case 4:
                Image1.ImageUrl = "~/Image/temp/fig5.png";
                break;
            case 5:
                Image1.ImageUrl = "~/Image/temp/fig6.png";
                break;
            default:
                break;
        }
        Session["simIPCC"] = RadioButtonList2.SelectedValue;    //# SPECIFY IPCC
SCENARIO //★★★
    }
    //************************************************************
    // DropDownList1:
    //      1. Select Mitigation pct and store in int[] var MitArr
```

```
    //      2. Validate Mitigation selection
    //      3. Update the Mitigation allocation cht
    //*************************************************************
    protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
    {
        //--------------------------
        // 1. UPDATE MitArr
        //--------------------------
        //int[] MitArr = new int[3];
        MitArr[0] =
Convert.ToInt32(Convert.ToDouble(DropDownList1.SelectedValue)*0.01*AnnBgt);
        MitArr[1] =
Convert.ToInt32(Convert.ToDouble(DropDownList2.SelectedValue)*0.01*AnnBgt);
        MitArr[2] =
Convert.ToInt32(Convert.ToDouble(DropDownList3.SelectedValue)*0.01*AnnBgt);
        //■■■
        lblAnnBgtChk.Text=Convert.ToString(AnnBgt);
        lblResA.Text = Convert.ToString(MitArr[0]);
        lblResB.Text = Convert.ToString(MitArr[1]);
        lblResC.Text = Convert.ToString(MitArr[2]);


        //---------------------------------------
        // 2. DATA VALIDATION (!warning message)
        //---------------------------------------
        int TotMit;
        TotMit = MitArr.Sum();

        if (TotMit > AnnBgt)
        {
            Label2.Attributes.Add("style", "color:Red");
            //Label2.Text = "You can only allocate 15% mitigation for all cities.";
            Label2.Text = "You don't have enough budget to use.";
            btnNext.Enabled = false;
        }
        else if (TotMit == AnnBgt)
        {
            Label2.Attributes.Add("style", "color:blue");
            Label2.Text = "You've used ALL the budget.";
            btnNext.Enabled = true;
        }
        else
        {
            Label2.Attributes.Add("style", "");
            //Label2.Text = "You still have extra resources to mitigate " +
Convert.ToString(AnnBgt - TotMit) +
            //              " % of damage.";
            Label2.Text = "You still have extra budget " + Convert.ToString(AnnBgt
- TotMit) + " million $ to use.";
            btnNext.Enabled = true;
        }
```

```
        //---------------------------------------------------
        // 3. UPDATING THE CHART (Showing value in percentage)
        //---------------------------------------------------
        Chart1.Series[0].Points[0].SetValueY(DropDownList1.SelectedValue);
        //Chart1.Series[0].Points[0].AxisLabel = "CityA";
        Chart1.Series[0].Points[1].SetValueY(DropDownList2.SelectedValue);
        //Chart1.Series[0].Points[1].AxisLabel = "CityB";
        Chart1.Series[0].Points[2].SetValueY(DropDownList3.SelectedValue);
        //Chart1.Series[0].Points[2].AxisLabel = "CityC";
        Chart1.Series[0].Points[3].SetValueY(100 -

Convert.ToDouble(DropDownList1.SelectedValue) -

Convert.ToDouble(DropDownList2.SelectedValue) -

Convert.ToDouble(DropDownList3.SelectedValue));
        //Chart1.Series[0].Points[3].AxisLabel = "Quota";

        //--------------------------
        // Store var in Session
        //--------------------------
        Session["MitArr"] = MitArr;       //★★★  Store Mitigation array as
dollars
    }


//*****************************************************************************
// Page_UnLoad:
//     1. Passing Selected Options (Period; IPCC sno)from results of Web Control
//        to Sessions
//     2. Connect to SnoDb and retrieve data
//*****************************************************************************
    protected void Page_Unload(object sender, EventArgs e)
    {
        //--------------------------------------------------------
        // 1.Passing Selected Options (from WebControl) to Sessions
        //---------------------------------------------------------
        //Session["simPrd"] = RadioButtonList1.SelectedValue;    //# SPECIFY
SIMULATION PERIOD  //★★★
        //Session["simIPCC"] = RadioButtonList2.SelectedValue;   //# SPECIFY IPCC
SCENARIO //★★★
        //Session["simYear"] = DateTime.Now.Year;              //# Ini simYear
using current year   //★★★

        //---------------------------------
        // 2. CONNECT TO SCENARIO DATABASE
        //---------------------------------
        using (OleDbConnection cn = new OleDbConnection())
        {
            // Set connection string
            //! Replace ACE with Jet
```

```
            //cn.ConnectionString = "Provider=Microsoft.ACE.OLEDB.12.0;" +
            //                    "Data Source=" +
Server.MapPath("~/App_Data/IPCCprj.accdb");
            cn.ConnectionString = "Provider=Microsoft.JET.OLEDB.4.0;" +
                                  "Data Source=" +
Server.MapPath("~/App_Data/IPCCprj.mdb");
            cn.Open();

            // DataSet
            DataSet ds = new DataSet();

            // DataAdapter
            OleDbDataAdapter daCO2 = new OleDbDataAdapter("SELECT * FROM CO2", cn);
            daCO2.Fill(ds, "CO2");
            OleDbDataAdapter daTmp = new OleDbDataAdapter("SELECT * FROM TmpInc",
cn);
            daTmp.Fill(ds, "TmpInc");

            // DataTable
            DataTable dtCO2, dtTmp;
            dtCO2 = ds.Tables["CO2"];
            dtTmp = ds.Tables["TmpInc"];

            // Var to retrieve Session value (from WC)
            simIPCC = Convert.ToInt32(Session["simIPCC"]);  //★★★

            // ArrayList
            ArrayList snroCO2arr = new ArrayList();
            ArrayList snroTmparr = new ArrayList();

            foreach (DataRow dtRow in dtCO2.Rows)
            {
                snroCO2arr.Add(dtRow[simIPCC].ToString());    //# Get the CO2
projection for specific scenario
            }

            foreach (DataRow dtRow in dtTmp.Rows)
            {
                snroTmparr.Add(dtRow[simIPCC].ToString());   //# Get the Tmp
projection for specific scenario
            }

            // Session vars to store data form DB
            Session["snroCO2arr"] = snroCO2arr;     //★★★
            Session["snroTmparr"] = snroTmparr;     //★★★
        }


        //------------------------------------------------------------------
        // CONNECT TO HURRICANE DAMAGE DATABASE: HurrMx.accdb
        //------------------------------------------------------------------
```

```
        using (OleDbConnection cn2 = new OleDbConnection())
        {
            // Set connection string
            //cn2.ConnectionString = "Provider=Microsoft.ACE.OLEDB.12.0;" +
            //                    "Data Source=" +
Server.MapPath("~/App_Data/HurrMx.accdb");
            cn2.ConnectionString = "Provider=Microsoft.JET.OLEDB.4.0;" +
                                  "Data Source=" +
Server.MapPath("~/App_Data/HurrMx.mdb");
            cn2.Open();

            DataSet ds2 = new DataSet();

            // DATAADAPTER
            OleDbDataAdapter daCityA = new OleDbDataAdapter("SELECT * FROM CityA",
cn2);
            daCityA.Fill(ds2, "CityA");
            OleDbDataAdapter daCityB = new OleDbDataAdapter("SELECT * FROM CityB",
cn2);
            daCityA.Fill(ds2, "CityB");
            OleDbDataAdapter daCityC = new OleDbDataAdapter("SELECT * FROM CityC",
cn2);
            daCityA.Fill(ds2, "CityC");

            // DATATABLE[]
            DataTable[] dtCity = new DataTable[3];
            dtCity[0] = ds2.Tables["CityA"];
            dtCity[1] = ds2.Tables["CityB"];
            dtCity[2] = ds2.Tables["CityC"];

            // Session to store data from DataTable[] dtCity
            Session["dtCity"] = dtCity;    //★★★
        }

        //---------------------------------------------------------------------
        // PASSING ARRAY MitArr (STORE MITIGATION PCT); From...
        //---------------------------------------------------------------------
        //Session["MitArr"] = MitArr;       //★★★
        Session["PopArr"] = PopArr;       //★★★

        //---------------------------------------------------------------------
        // PASSING ARRAYLIST (STORE SERIES DATA)     ; From...
        //---------------------------------------------------------------------
        Session["DmgSrsA"] = DmgSrsA;    //★★★       // Series Dmg
        Session["DmgSrsB"] = DmgSrsB;    //★★★       // Series Dmg
        Session["DmgSrsC"] = DmgSrsC;    //★★★       // Series Dmg
        Session["DmgSrsAr"] = DmgSrsAr;    //★★★      // Series Dmg reduced
        Session["DmgSrsBr"] = DmgSrsBr;    //★★★      // Series Dmg reduced
        Session["DmgSrsCr"] = DmgSrsCr;    //★★★      // Series Dmg reduced

        Session["PopSrsA"] = PopSrsA;    //★★★       // Series Pop
```

```
        Session["PopSrsB"] = PopSrsB;    //★★★       // Series Pop
        Session["PopSrsC"] = PopSrsC;    //★★★       // Series Pop
        Session["PopSrsAr"] = PopSrsAr;    //★★★     // Series Pop reduced
        Session["PopSrsBr"] = PopSrsBr;    //★★★     // Series Pop reduced
        Session["PopSrsCr"] = PopSrsCr;    //★★★     // Series Pop reduced

        Session["AraSrsA"] = AraSrsA;    //★★★       // Series Ara
        Session["AraSrsB"] = AraSrsB;    //★★★       // Series Ara
        Session["AraSrsC"] = AraSrsC;    //★★★       // Series Ara

        // Passing Initial CpT
        Session["CpTSrs"] = CpTSrs;    //★★★
        Session["CpRSrs"] = CpRSrs;    //★★★
    }

    //=======================================================
    // User Defined Method
    //=======================================================

    private double getBudget()
    {
        int seed = (int)DateTime.Now.Ticks;
        Random r = new Random(seed);
        double annbgt = r.Next(8, 11)*10;
        return annbgt;
    }

    //=======================================================
    // GO TO NEXT PAGE
    //=======================================================
    protected void btnNext_Click(object sender, EventArgs e)
    {
        //Response.Redirect("~/Page06.aspx");
        Response.Redirect("~/ModSlt.aspx");
    }
}
```

H.6. ModSlt.aspx

```
//=================================================================
// Purpose:
//     Allow the user to select whether to run the simulation automatically or
manually
//=================================================================
//--Default
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
public partial class ModSlt : System.Web.UI.Page
{
   //===========================================================
   //  Page Events
   //===========================================================
   protected void Page_Init(object sender, EventArgs e)
   {
      //Panel1.Visible = Convert.ToBoolean(Session["showMonitor"]);
      Panel1.Visible = Convert.ToBoolean(0);
   }
   protected void Page_Load(object sender, EventArgs e)
   {

   }
   protected void Page_Unload(object sender, EventArgs e)
   {

   }
   //===========================================================
   //  Web Control Events
   //===========================================================
   protected void Button1_Click(object sender, EventArgs e)
   {
      Session["btnVis"] = false;
      Session["lblVis"] = false;
      Session["tmrEab"] = true;
      Response.Redirect("~/Page06.aspx");
   }
   protected void Button2_Click(object sender, EventArgs e)
   {
      Session["btnVis"] = true;
      Session["lblVis"] = true;
      Session["tmrEab"] = false;
      Response.Redirect("~/Page06.aspx");
   }
}
```

H.7. Page06.aspx

```
//==============================================================================
//  Purpose
//      1. Simulate hurricane events
//  Self-defined Function
//      1. Sim()
//      2. bool getOcc()           # Simulate occurance
//      3. int getLfl()            # Landfall location
//      4. double getCpT(double)    # Hurricane central pressure (theoretical)
//      5. double getCpR(double)    # Hurricane central pressure (real)
//      6. int getHurCat(double)    # Hurricane Category
//      7. double getRp()           # Hurricane Radius to the maximum wind
//      8. int getSnoLn()           # Scenario Line in lookup table, finds the damage
value
//==============================================================================
//--Default
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
//--ADDED--
using System.Collections;
using System.Data;
using System.Data.OleDb;

public partial class Default2 : System.Web.UI.Page
{

//==============================================================================
    // Using Random Numbers

//==============================================================================
    Random Rnd = new Random();


//==============================================================================
    // VARIABLES

//==============================================================================

//##############################################################################
    static string[] CityNames = new string[3];//{"CityA","CityB","CityC"};
    static DataTable dtHurNam=new DataTable();


//##############################################################################
    //static double CpT=new double();
```

```
    static int[] MitArr = new int[3];    //--RETRIVE MitArr
    static int[] PopArr = new int[3];

    static ArrayList snroCO2arr = new ArrayList();
    static ArrayList snroTmparr = new ArrayList();

    static int simPrd = new int();
    static int simIPCC = new int();
    static int simYear = new int();     // GAME YEAR
    //static int nowYear = new int();     // 2013

    static int prdStartYear = new int();    // Period start year
    static int prdEndYear = new int();     // period end year

    static DataTable[] dtCity = new DataTable[3];   // Dmg table for ABC
    static DataTable dtCityLF = new DataTable();    // Dmg table for landfall city

    static double[] DmgPrdT = new double[3];      // Period Dmg
    static double[] PopPrdT = new double[3];      // Period Dmg
    static double[] AraPrdT = new double[3];      // Period Dmg

    static double[] DmgPrdR = new double[3];      // Period Dmg reduced
    static double[] PopPrdR = new double[3];      // Period Dmg reduced
    static double[] AraPrdR = new double[3];      // Period Dmg reduced


//###############################################################################
    static ArrayList DmgSrsA = new ArrayList(); // Series Dmg
    static ArrayList DmgSrsB = new ArrayList(); // Series Dmg
    static ArrayList DmgSrsC = new ArrayList(); // Series Dmg
    static ArrayList DmgSrsAr = new ArrayList(); // Series Dmg reduced
    static ArrayList DmgSrsBr = new ArrayList(); // Series Dmg reduced
    static ArrayList DmgSrsCr = new ArrayList(); // Series Dmg reduced

    static ArrayList PopSrsA = new ArrayList(); // Series Pop
    static ArrayList PopSrsB = new ArrayList(); // Series Pop
    static ArrayList PopSrsC = new ArrayList(); // Series Pop
    static ArrayList PopSrsAr = new ArrayList(); // Series Pop reduced
    static ArrayList PopSrsBr = new ArrayList(); // Series Pop reduced
    static ArrayList PopSrsCr = new ArrayList(); // Series Pop reduced

    static ArrayList AraSrsA = new ArrayList(); // Series Ara
    static ArrayList AraSrsB = new ArrayList(); // Series Ara
    static ArrayList AraSrsC = new ArrayList(); // Series Ara

    static ArrayList CpTSrs = new ArrayList(); // Series Srs Theoretical Cp
    static ArrayList CpRSrs = new ArrayList(); // Series Srs True Cp


//###############################################################################
    static double AnnBgt;      // Global variable Annual Budget
```

```
//============================================
// Page Events
//============================================
//********************************************
// Page Init
//********************************************
protected void Page_Init(object sender, EventArgs e)
{
   Panel1.Visible = Convert.ToBoolean(Session["showMonitor"]);
}
//********************************************
// Page Load
//********************************************
protected void Page_Load(object sender, EventArgs e)
{
   if (!IsPostBack)
   {
      //----------------------------------------
      // Retrieve CityNames/Hurricane Names
      //----------------------------------------
      CityNames = (string[])Session["CityNames"]; //★★★
      dtHurNam = (DataTable)Session["dtHurNam"];  //★★★
      AnnBgt = (double)Session["AnnBgt"];        //★★★

      //----------------------------------------
      // Clear DmgPrd Array
      //----------------------------------------
      Array.Clear(DmgPrdT, 0, DmgPrdT.Length);
      Array.Clear(PopPrdT, 0, PopPrdT.Length);
      Array.Clear(AraPrdT, 0, AraPrdT.Length);

      Array.Clear(DmgPrdR, 0, DmgPrdR.Length);
      Array.Clear(PopPrdR, 0, PopPrdR.Length);
      Array.Clear(AraPrdR, 0, AraPrdR.Length);

      // Obtain Initial CpT
      // CpT = (double)Session["CpT"];

      //--------------------------------------
      // From Simulation Mode (Auto/Manual)
      //--------------------------------------
      Timer1.Enabled=(bool)Session["tmrEab"];
      Button1.Visible=(bool)Session["btnVis"];
      lblNextYr.Text = "Click to proceed";
      lblNextYr.Visible = (bool)Session["lblVis"];

      //------------------------------------------
      // Default Control Properties
      //------------------------------------------
      //--DEFAULT btnNext CONDITION
```

```
            btnNext.Enabled = false;
            lblNextRun.Visible = false;

            //----------------------------------------
            // RETRIEVE VARIABLES FROM SESSIONS (From Page05/07)
            //----------------------------------------
            simYear = Convert.ToInt32(Session["simYear"]);
            simIPCC = Convert.ToInt32(Session["simIPCC"]);
            simPrd = Convert.ToInt32(Session["simPrd"]);

            //----------------------------------------
            // Update MitArr and PopArr        (From Page05/07)
            //----------------------------------------
            MitArr = (int[])Session["MitArr"];            //--UPDATE EACH PERIOD
            PopArr = (int[])Session["PopArr"];            //--UPDATE EACH PERIOD

            //# check
            lblMit0.Text = MitArr[0].ToString();
            lblMit1.Text = MitArr[1].ToString();
            lblMit2.Text = MitArr[2].ToString();

            //----------------------------------------
            // RETRIEVE Scenario CO2 and TmpInc (From Page05/07)
            //----------------------------------------
            snroCO2arr = (ArrayList)Session["snroCO2arr"];
            snroTmparr = (ArrayList)Session["snroTmparr"];

            //--OBTAIN SELECTED IPCC SNRO DATA AND DISPLAY
            lblSimYear.Text = simYear.ToString();


            //# CO2: Set current CO2 as the latest value
            int curYr = (int)Session["nowYear"];
            double dltCO2 = Convert.ToDouble(snroCO2arr[curYr - 2000]) -
(double)Session["CurCO2"];
            int corCO2 = Convert.ToInt32(Convert.ToDouble(snroCO2arr[simYear - 2000])
- dltCO2);
            lblSimCO2.Text = corCO2.ToString();
            ////lblSimCO2.Text = snroCO2arr[simYear - 2000].ToString();

            //# Temp: Set current Temp as the latest value
            double cordltTmp = Convert.ToDouble(snroTmparr[simYear - 2000]) -
Convert.ToDouble(snroTmparr[curYr - 2000]);
            lblSimTmp.Text = cordltTmp.ToString();
            ////lblSimTmp.Text = snroTmparr[simYear - 2000].ToString();

            lblCurYr.Text = Convert.ToString(Session["nowYear"]);
            //lblYrNow.Text = nowYear.ToString();
            //lblYrNow.Text = DateTime.Now.Year.ToString();

            //------------------------------------------
```

```
//  Retrieve dtHurrMx from previous session
//------------------------------------------
dtCity=(DataTable[])Session["dtCity"];   //# From Page05

//---------------------------------------------
// ArrayList to keep records; Update?
//---------------------------------------------
DmgSrsA = (ArrayList)Session["DmgSrsA"];    //★★★
DmgSrsB = (ArrayList)Session["DmgSrsB"];    //★★★
DmgSrsC = (ArrayList)Session["DmgSrsC"];    //★★★
DmgSrsAr = (ArrayList)Session["DmgSrsAr"];    //★★★
DmgSrsBr = (ArrayList)Session["DmgSrsBr"];    //★★★
DmgSrsCr = (ArrayList)Session["DmgSrsCr"];    //★★★

PopSrsA = (ArrayList)Session["PopSrsA"];    //★★★
PopSrsB = (ArrayList)Session["PopSrsB"];    //★★★
PopSrsC = (ArrayList)Session["PopSrsC"];    //★★★
PopSrsAr = (ArrayList)Session["PopSrsAr"];    //★★★
PopSrsBr = (ArrayList)Session["PopSrsBr"];    //★★★
PopSrsCr = (ArrayList)Session["PopSrsCr"];    //★★★

AraSrsA = (ArrayList)Session["AraSrsA"];    //★★★
AraSrsB = (ArrayList)Session["AraSrsB"];    //★★★
AraSrsC = (ArrayList)Session["AraSrsC"];    //★★★

CpTSrs = (ArrayList)Session["CpTSrs"];     //★★★
CpRSrs = (ArrayList)Session["CpRSrs"];     //★★★

//-----------------------------------------------
// Define StartYear and EndYear of this Period
//-----------------------------------------------
prdStartYear = Convert.ToInt32(Session["simYear"]);        //--UPDATE
EACH TICK
prdEndYear = prdStartYear + simPrd;           //# moved from timer
prdEndYear = (prdEndYear > 2100) ? 2100 : prdEndYear;

//-----------------------------
// OBTAIN nowYear VAR (why?)
//-----------------------------
//nowYear = DateTime.Now.Year;
//Session["nowYear"] = nowYear;

//---------------
// Run Simulation
//---------------
Sim();

//# Check
//lblDmgAllACnt.Text = DmgSrsA.Count.ToString();
//lblDmgAllBCnt.Text = DmgSrsB.Count.ToString();
//lblDmgAllCCnt.Text = DmgSrsC.Count.ToString();
```

```
            //lblPopRACnt.Text = PopSrsA.Count.ToString();
            //lblPopRBCnt.Text = PopSrsB.Count.ToString();
            //lblPopRCCnt.Text = PopSrsC.Count.ToString();
        }
    }

    //*********************************************
    // Page Unload
    //*********************************************
    protected void Page_Unload(object sender, EventArgs e)
    {

    }

    //=============================================
    // Web Control Events
    //=============================================
    //*********************************************
    // Timer1_Tick
    //*********************************************
    protected void Timer1_Tick(object sender, EventArgs e)
    {
        //--DEFINE THE END OF SIM PRD (check line 93)
        //prdEndYear = prdStartYear + simPrd;
        //prdEndYear = (prdEndYear > 2100) ? 2100 : prdEndYear;

        //--UPDATE(THEN DISPLAY) CO2 AND TMP EACH SIM YEAR
        Session["simYear"] = Convert.ToInt32(Session["simYear"]) + 1;  // UPDATE SIM
YEAR
        simYear = (int)Session["simYear"];
        lblSimYear.Text = Convert.ToString(Session["simYear"]);       // DISPLAY

        //@@ lblSimCO2.Text = snroCO2arr[simYear - 2000].ToString();       // OBTAIN
CO2 VALUE
        int curYr = (int)Session["nowYear"];
        double dltCO2 = Convert.ToDouble(snroCO2arr[curYr - 2000]) -
(double)Session["CurCO2"];
        int corCO2 = Convert.ToInt32(Convert.ToDouble(snroCO2arr[simYear - 2000])
- dltCO2);
        lblSimCO2.Text = corCO2.ToString();

        //@@ lblSimTmp.Text = snroTmparr[simYear - 2000].ToString();       // OBTAIN
TMP VALUE
        double cordltTmp = Convert.ToDouble(snroTmparr[simYear - 2000]) -
Convert.ToDouble(snroTmparr[curYr - 2000]);
        lblSimTmp.Text = cordltTmp.ToString();

        //# check
        lblstart.Text = prdStartYear.ToString();
        lblEnd.Text = prdEndYear.ToString();
```

```
        //-------------
        // SIMULATION
        //-------------
        Sim();

        //--WHEN REACHING THE PERIOD END, STOP THE TIMER AND ENABLE btnNext
        if (simYear == prdEndYear)
        {
            //Session["simYear"] = Convert.ToInt32(Session["simYear"]) + 1;   //
UPDATE Session["simYear"];
            Timer1.Enabled = false;     // STOP THE TIMER
            btnNext.Enabled = true;     // ENABLE THE btnNext BUTTON
            lblNextRun.Visible = true;  //
        }
    }
    //**********************************************
    // Button1_Click
    //**********************************************
    protected void Button1_Click(object sender, EventArgs e)
    {
        //--UPDATE(THEN DISPLAY) CO2 AND TMP EACH SIM YEAR
        Session["simYear"] = Convert.ToInt32(Session["simYear"]) + 1;  // UPDATE SIM
YEAR
        simYear = (int)Session["simYear"];
        lblSimYear.Text = Convert.ToString(Session["simYear"]);       // DISPLAY

        //@@ lblSimCO2.Text = snroCO2arr[simYear - 2000].ToString();       // OBTAIN
CO2 VALUE
        int curYr = (int)Session["nowYear"];
        double dltCO2 = Convert.ToDouble(snroCO2arr[curYr - 2000]) -
(double)Session["CurCO2"];
        int corCO2 = Convert.ToInt32(Convert.ToDouble(snroCO2arr[simYear - 2000])
- dltCO2);
        lblSimCO2.Text = corCO2.ToString();

        //@@ lblSimTmp.Text = snroTmparr[simYear - 2000].ToString();       // OBTAIN
TMP VALUE
        double cordltTmp = Convert.ToDouble(snroTmparr[simYear - 2000]) -
Convert.ToDouble(snroTmparr[curYr - 2000]);
        lblSimTmp.Text = cordltTmp.ToString();

        //# check
        lblstart.Text = prdStartYear.ToString();
        lblEnd.Text = prdEndYear.ToString();

        //-------------
        // SIMULATION
        //-------------
        Sim();

        //--WHEN REACHING THE PERIOD END, STOP THE TIMER AND ENABLE btnNext
```

```
        if (simYear == prdEndYear)
        {
            Button1.Visible = false;
            //lblNextYr.Text = "Click Next at the bottom.";
            lblNextYr.Visible = false;
            //lblNextBtn.Text = "Click Next at the bottom";
            btnNext.Enabled = true;
            lblNextRun.Visible = true;  //
        }
    }

    //==================================================
    // User-Defined Function
    //==================================================
    //*********************************************
    // Sim()
    //*********************************************
    private void Sim()
    {
        //----------------------------------------------------------------
        // Variables
        //----------------------------------------------------------------
        bool Occ = new bool();         //# OCCURANCE

        int Lfl = new int();           //# LANDFALL LOCATION
        int SnoLn = new int();         //# SCENARIO NUMBER
        int HurCat = new int();        // # HURRICANE CATEGORY

        double Tmp = new double();      //# TEMPERATURE
        double CpT = new double();      //# THEORETICAL CENTRAL PRESSURE
        double CpR = new double();      //# REAL CENTRAL PRESSURE
        double Rp = new double();       //# RADIUS

        double[] DmgT = new double[3];  //# THEORETICAL DAMAGE
        double[] PopT = new double[3];  //# THEORETICAL DAMAGE
        double[] AraT = new double[3];  //# THEORETICAL DAMAGE

        double[] DmgR = new double[3];  //# REAL DAMAGE
        double[] PopR = new double[3];  //# REAL DAMAGE
        double[] AraR = new double[3];  //# REAL DAMAGE
        //-----------------------------------------------
        // Simulating Occurance
        //-----------------------------------------------
        Occ = getOcc();     //# Get Occurance
        //Tmp = Convert.ToDouble(snroTmparr[simYear - 2000]);
        int curYr = (int)Session["nowYear"];
        Tmp = Convert.ToDouble(snroTmparr[simYear - 2000]) -
Convert.ToDouble(snroTmparr[curYr - 2000]);

        //--Sim/Store Theroical Cp
        CpT = getCpT(Tmp);   //# Get CpT
```

```
        CpTSrs.Add(CpT);     //# Store CpT ★★★

        //██ check
        lblOcc.Text = Occ.ToString();
        lblTmp.Text = Tmp.ToString();
        lblCpT.Text = CpT.ToString();

        if (Occ)  // IF HURRICANE OCCURS
        {
            //# USE RANDOM
            //int seed = (int)DateTime.Now.Ticks;
            //Random Rnd = new Random(seed);
            //Random Rnd = new Random();

            //*********Get Hurricane Name and Characteristics******
            string HurName = dtHurNam.Rows[Rnd.Next(0, 20)][Rnd.Next(2,
7)].ToString();

            //--Sim/Store Real Cp
            CpR = getCpR(CpT);         //# Get CpR
            lblCpR.Text = CpR.ToString();
            CpRSrs.Add(CpR);           //★★★

            //*********Hurricane Info************
            lblHurName.Text = HurName;
            //lblHurCat.Text = "";
            lblHurCp.Text = Math.Floor(CpR).ToString();
            //lblHurWs.Text = "";
            MultiView3.SetActiveView(ViwOcc1);


            //--Sim Category ; function of Cp?
            HurCat = getHurCat(CpR);
            lblHurCat.Text = HurCat.ToString();

            //--Sim Rp
            Rp = getRp();
            lblRp.Text = Rp.ToString();

            //--Sim SnoLn (Scenario line in damage table)
            //SnoLn = 0;
            //lblSnoLn.Text = SnoLn.ToString();

            //--Simulating Landfall location
            Lfl = getLfl();
            lblLfl.Text = Lfl.ToString();

            switch (Lfl) // IF 0 OR 4, NO DMG TO CITIES
            {
                case 0:
                    //***********Hurricane Description*************
```

```
                    //(1) Hurricane Track Image
                    ImgTrack.ImageUrl="./Image/hurricane/LF0.gif";
                    //(2) Image
                    ImgFlood.ImageUrl = "./Image/FloodOut/fld1.jpg";
                    MultiView1.SetActiveView(ViewFlood);
                    //(3) Hurricane Info
                       //# outside the loop
                    //(4) Description
                    lblHurNamLF0.Text=HurName;
                    lblLandfall0.Text =
Rnd.Next(10,20).ToString();//Lfl.ToString();
                    lblNLF0.Text = CityNames[0];
                    //MultiView2.SetActiveView(ViwOccLf0);
                    MultiView4.SetActiveView(ViewOccLf0);
                    //***********Time/CO2/Temperature*************
                    //#check
                    lblMsg.Text = "Made landfall outside cities.";

                    //--Sim Rp
                    //Rp = getRp();
                    //lblRp.Text = Rp.ToString();

                    //--Sim SnoLn
                    //SnoLn = 0;
                    //lblSnoLn.Text = SnoLn.ToString();

                    //--Sim Dmg (No Dmg to Cities, clear Dmg Mx for "this" year)
                    Array.Clear(DmgT, 0, DmgT.Length);
                    Array.Clear(PopT, 0, PopT.Length);
                    Array.Clear(AraT, 0, AraT.Length);

                    Array.Clear(DmgR, 0, DmgR.Length);
                    Array.Clear(PopR, 0, PopR.Length);
                    break;
                case 4:
                    //***********Hurricane Description*************
                    //(1) Hurricane Track Image
                    ImgTrack.ImageUrl = "./Image/hurricane/LF4.gif";
                    //(2) Image
                    ImgFlood.ImageUrl = "./Image/FloodOut/fld1.jpg";
                    MultiView1.SetActiveView(ViewFlood);
                    //(3) Hurricane Info
                       //# outside the loop
                    //(4) Description
                    lblHurNamLF4.Text = HurName;
                    lblLandfall4.Text =
Rnd.Next(10,20).ToString();//Lfl.ToString();
                    lblNLF4.Text = CityNames[2];
                    //MultiView2.SetActiveView(ViwOccLf4);
                    MultiView4.SetActiveView(ViewOccLf4);
```

```
            //***********Time/CO2/Temperature**************
            //■■ check
            lblMsg.Text = "Made landfall outside cities.";

            //--Sim Rp
            //Rp = getRp();
            //lblRp.Text = Rp.ToString();

            //--Sim SnoLn
            //SnoLn = 0;
            //lblSnoLn.Text = SnoLn.ToString();

            //--Sim Dmg (No Dmg to Cities, clear Dmg Mx for "this" year)
            Array.Clear(DmgT, 0, DmgT.Length);
            Array.Clear(PopT, 0, PopT.Length);
            Array.Clear(AraT, 0, AraT.Length);

            Array.Clear(DmgR, 0, DmgR.Length);
            Array.Clear(PopR, 0, PopR.Length);

            //# check
            lblDmgTA.Text = DmgT[0].ToString();
            lblDmgTB.Text = DmgT[1].ToString();
            lblDmgTC.Text = DmgT[2].ToString();
            lblPopTA.Text = PopT[0].ToString();
            lblPopTB.Text = PopT[1].ToString();
            lblPopTC.Text = PopT[2].ToString();

            break;

        // # if case 1~3 make landfall at cities
        case 1:
        case 2:
        case 3:
            //***********Hurricane Description**************
            //(1) Hurricane Track Image
            ImgTrack.ImageUrl = "./Image/hurricane/LF"+Lfl.ToString()
+".gif";

            //(2) Image
            ImgFlood.ImageUrl = "./Image/FloodIn/fld1.jpg";
            MultiView1.SetActiveView(ViewFlood);
            //(3) Hurricane Info
                //# Outside the loop
            //(4) Description (HurName; CityName; View; Damage)
            lblHurNamLF123.Text = HurName;
            lblLF123.Text = CityNames[Lfl-1];//Lfl.ToString();
            //MultiView2.SetActiveView(ViwOccLf123);
            MultiView4.SetActiveView(ViewOccLf123);

            //■■ Check
            lblMsg.Text = "Made landfall.";
```

```
//--Obtaining the dmg data that hurricane made landfall
dtCityLF = dtCity[Lfl - 1];    //◆◆◆

//--Sim Rp
//Rp = getRp();
//lblRp.Text = Rp.ToString();

//--Sim SnoLn (Only when the hurrican made landfall at 3 cities)
SnoLn = getSnoLn(CpR, Rp);
lblSnoLn.Text = SnoLn.ToString();

//--------------------------------
//--Sim Dmg (obtain from dtCityLF)
//--------------------------------
DmgT[Lfl - 1] = Convert.ToDouble(dtCityLF.Rows[SnoLn - 1][3]);
PopT[Lfl - 1] = Convert.ToDouble(dtCityLF.Rows[SnoLn - 1][4]);
AraT[Lfl - 1] = Convert.ToDouble(dtCityLF.Rows[SnoLn - 1][5]);

////DmgR[Lfl - 1] = Math.Floor(DmgT[Lfl - 1] * (1 - 0.01 * MitArr[Lfl
- 1]));

////PopR[Lfl - 1] = Math.Floor(PopT[Lfl - 1] * (1 - 0.01 * PopArr[Lfl
- 1]));

// - SrgDmg(DmgT) and Insured(DmgR)
if (DmgT[Lfl - 1] >= MitArr[Lfl - 1]*0.8)
{
   DmgR[Lfl - 1] = MitArr[Lfl - 1]*0.8;
}
else
{
   DmgR[Lfl - 1] = DmgT[Lfl - 1];
}
// - PopDmg(PopT) and Secured(PopR)
if (PopT[Lfl - 1] >= MitArr[Lfl - 1]*15)
{
   PopR[Lfl - 1] = MitArr[Lfl - 1]*15;
}
else
{
   PopR[Lfl - 1] = PopT[Lfl - 1];
}
//--------------------------------
//# Display damage
//--------------------------------
//# Showing the hurricane damage
lblDmg.Text = Convert.ToString(dtCityLF.Rows[SnoLn - 1][3]);
lblPop.Text = Convert.ToString(dtCityLF.Rows[SnoLn - 1][4]);
lblAra.Text = Convert.ToString(dtCityLF.Rows[SnoLn - 1][5]);
//# Showing the insured amount and secured population
lblInsured.Text = Convert.ToString(DmgR[Lfl - 1]);
```

```
            lblSecured.Text = Convert.ToString(PopR[Lfl - 1]);

            // - Add Information

            //■■ check
            lblDmgTA.Text = DmgT[0].ToString();
            lblDmgTB.Text = DmgT[1].ToString();
            lblDmgTC.Text = DmgT[2].ToString();
            lblPopTA.Text = PopT[0].ToString();
            lblPopTB.Text = PopT[1].ToString();
            lblPopTC.Text = PopT[2].ToString();
            break;
        default:
            break;
    }
}
else //# If hurricane doesn't strike the coastline
{
    //# Use Random
    //Random Rnd = new Random();

    //*********Hurricane Info************
    MultiView3.SetActiveView(ViwOcc0);

    //***********Hurricane Description**************
    //(1) Hurricane Track Image
    ImgTrack.ImageUrl = "./Image/hurricane/LF5.png";
    //(2) Image
    ImgSunny.ImageUrl = "./Image/Sunny/Sunny" + Rnd.Next(1, 7) + ".jpg";
    MultiView1.SetActiveView(ViewSunny);
    //(3) Hurricane Info
    //MultiView2.SetActiveView(ViwNoOcc);
    MultiView4.SetActiveView(ViewNoOcc);
    //(4) Description

    //--Sim/Store Real Cp
    CpRSrs.Add(null);

    lblMsg.Text = "No hurricane strikes the coastline.";
    //------------------------
    // CLEAR ALL VARIABLES
    //------------------------
    //--Sim Landfall location
    Lfl = -999;
    lblLfl.Text = Lfl.ToString();

    //--Sim Cp=f(Tmp)
    //CpT = 0;
    //CpR = 0;
    //lblCpR.Text = CpR.ToString();
```

```
    //--Sim Rp
    Rp = -999;
    lblRp.Text = Rp.ToString();

    //--Sim SnoLn
    SnoLn = -999;
    lblSnoLn.Text = SnoLn.ToString();

    //--Sim Dmg (No Hurricane, No damage)
    Array.Clear(DmgT, 0, DmgT.Length);
    Array.Clear(PopT, 0, PopT.Length);
    Array.Clear(AraT, 0, AraT.Length);

    Array.Clear(DmgR, 0, DmgR.Length);
    Array.Clear(PopR, 0, PopR.Length);
    Array.Clear(AraR, 0, AraR.Length);

    //■■ Check
    lblDmgTA.Text = DmgT[0].ToString();
    lblDmgTB.Text = DmgT[1].ToString();
    lblDmgTC.Text = DmgT[2].ToString();
    lblPopTA.Text = PopT[0].ToString();
    lblPopTB.Text = PopT[1].ToString();
    lblPopTC.Text = PopT[2].ToString();
}
//------------------------------------------------------------------
// Calculate total Dmg for this period
//------------------------------------------------------------------
for (int i = 0; i < DmgPrdT.Length; i++)
{
    DmgPrdT[i] += DmgT[i];
    PopPrdT[i] += PopT[i];
    AraPrdT[i] += AraT[i];

    DmgPrdR[i] += DmgR[i];
    PopPrdR[i] += PopR[i];

    lblDmgPrdA.Text = DmgPrdT[0].ToString();
    lblDmgPrdB.Text = DmgPrdT[1].ToString();
    lblDmgPrdC.Text = DmgPrdT[2].ToString();
}
//Chart1.Series[0].Points.DataBindY(DmgPrd);
Session["DmgPrdT"] = DmgPrdT;   //★★★
Session["PopPrdT"] = PopPrdT;   //★★★
Session["AraPrdT"] = AraPrdT;   //★★★

Session["DmgPrdR"] = DmgPrdR;   //★★★
Session["PopPrdR"] = PopPrdR;   //★★★

//-----------------------------------------
// Update and Store data for whole series
```

```
//---------------------------------------
DmgSrsA.Add(DmgT[0]);         //# Actual Damage
DmgSrsB.Add(DmgT[1]);
DmgSrsC.Add(DmgT[2]);

DmgSrsAr.Add(DmgR[0]);        //# Insured Value
DmgSrsBr.Add(DmgR[1]);
DmgSrsCr.Add(DmgR[2]);

//DmgSrsAr.Add(DmgT[0] * (1 - 0.01 * MitArr[0]));
//DmgSrsBr.Add(DmgT[1] * (1 - 0.01 * MitArr[1]));
//DmgSrsCr.Add(DmgT[2] * (1 - 0.01 * MitArr[2]));

PopSrsA.Add(PopT[0]);         //# Actual Damagse
PopSrsB.Add(PopT[1]);
PopSrsC.Add(PopT[2]);

PopSrsAr.Add(PopR[0]);         //# Insured Value
PopSrsBr.Add(PopR[1]);
PopSrsCr.Add(PopR[2]);

//PopSrsAr.Add(PopT[0] * (1 - 0.01 * PopArr[0]));
//PopSrsBr.Add(PopT[1] * (1 - 0.01 * PopArr[1]));
//PopSrsCr.Add(PopT[2] * (1 - 0.01 * PopArr[2]));

AraSrsA.Add(AraT[0]);
AraSrsB.Add(AraT[1]);
AraSrsC.Add(AraT[2]);

//■■ Check
lblDmgAllACnt.Text = DmgSrsAr.Count.ToString();
lblDmgAllBCnt.Text = DmgSrsBr.Count.ToString();
lblDmgAllCCnt.Text = DmgSrsCr.Count.ToString();
lblPopRACnt.Text = AraSrsA.Count.ToString();
lblPopRBCnt.Text = AraSrsB.Count.ToString();
lblPopRCCnt.Text = AraSrsC.Count.ToString();

//-Store Session
Session["DmgSrsA"] = DmgSrsA;   //★★★
Session["DmgSrsB"] = DmgSrsB;   //★★★
Session["DmgSrsC"] = DmgSrsC;   //★★★
Session["DmgSrsAr"] = DmgSrsAr;   //★★★
Session["DmgSrsBr"] = DmgSrsBr;   //★★★
Session["DmgSrsCr"] = DmgSrsCr;   //★★★

Session["PopSrsA"] = PopSrsA;   //★★★
Session["PopSrsB"] = PopSrsB;   //★★★
Session["PopSrsC"] = PopSrsC;   //★★★
Session["PopSrsAr"] = PopSrsAr;   //★★★
Session["PopSrsBr"] = PopSrsBr;   //★★★
Session["PopSrsCr"] = PopSrsCr;   //★★★
```

```
    Session["AraSrsA"] = AraSrsA;   //★★★
    Session["AraSrsB"] = AraSrsB;   //★★★
    Session["AraSrsC"] = AraSrsC;   //★★★
} //# End Sim()

//**************************************************************************
// Define rate of occurance/ rate for a hurricane to strike the coastline
//**************************************************************************
private bool getOcc()
{
    //# Use random seed
    //int seed = (int)DateTime.Now.Ticks;
    //Random Rnd=new Random(seed);
    //Random Rnd = new Random();

    if (Rnd.NextDouble() < 0.5)
    {
        return true;
    }
    else
    {
        return false;
    }
}
//***************************************************
// Define the landfall loaction that hurricane makes
//***************************************************
private int getLfl()
{
    //# Use random seed
    //int seed = (int)DateTime.Now.Ticks;
    //Random Rnd = new Random(seed);
    //Random Rnd = new Random();

    return Rnd.Next(5);
}
//**************************************************
//-- testing
//**************************************************
//private double getTmp()
//{
//    //# Use random seed
//    //int seed = (int)DateTime.Now.Ticks;
//    //Random Rnd = new Random(seed);

//    return Rnd.Next(10, 30);
//}
//**************************************************
//  Get mean central pressure
//**************************************************
```

```
private double getCpT(double tmp)
{
   return 945-0.08*tmp*(1013-945);
}
//*************************************************
// Get real hurricane central pressure
//*************************************************
private double getCpR(double cpt)
{
   //# Use random seed
   //int seed = (int)DateTime.Now.Ticks;
   //Random Rnd = new Random(seed);
   //Random Rnd = new Random();

   return cpt+(Rnd.NextDouble()-0.5)*2*25;
}
//*************************************************
// Get hurricane category based on central pressure
//*************************************************
private int getHurCat(double cpr)
{
   int hurcat = new int();
   //return cpt + (Rnd.NextDouble() - 0.5) * 2 * 20;
   if (cpr >= 980)
   {
      hurcat = 1;
   }
   else if (cpr >= 965 && cpr <= 979)
   {
      hurcat = 2;
   }
   else if (cpr >= 945 && cpr <= 964)
   {
      hurcat = 3;
   }
   else if (cpr >= 920 && cpr <= 944)
   {
      hurcat = 4;
   }
   else
   {
      hurcat = 5;
   }
   return hurcat;
}
//*************************************************
// Get hurricane radius
//*************************************************
private double getRp()
{
   //# Use random seed
```

```
        //int seed = (int)DateTime.Now.Ticks;
        //Random Rnd = new Random(seed);
        //Random Rnd = new Random();

        return Math.Floor(4 + Rnd.NextDouble() * 60);
    }
    //*********************************************************************
    // Get scenario line number in the table (for looking up damage value)
    //*********************************************************************
    private int getSnoLn(double cpr, double rp)
    {
        int CpIdx = Convert.ToInt32(Math.Round((cpr - 770) / 20) + 1);
        int RpIdx = Convert.ToInt32(Math.Round((rp - 4) / 4) + 1);
        // FIND THE NUMBER OF LINE
        int snoln = (CpIdx - 1) * 16 + RpIdx;
        return snoln;
    }

    //=============================================
    // Go To Next Page
    //=============================================
    protected void btnNext_Click(object sender, EventArgs e)
    {
        // Passing CpT to next period
        // Session["CpT"] = CpT;

        //-- WHEN REACHING 2100 YR, GO TO FINAL PAGE
        if (simYear == 2100)
        {
            Response.Redirect("~/Final.aspx");
        }
        else
        {
            Response.Redirect("~/Page07.aspx");
        }
    }
}
```

H.8. Page07.aspx

```
//============================================================================
//  Purpose:
//      1. Showing Period Results
//      2. Update Resources
//============================================================================
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
//--ADDED
using System.Collections;
using System.Web.UI.DataVisualization.Charting;
using System.Drawing;

public partial class Default2 : System.Web.UI.Page
{
    //=======================================================
    // PUBLIC VARIABLES
    //=======================================================

//############################################################################
    //--VARS TO RETRIEVE SESSION DATA
    static int simYear=new int();

    static ArrayList snroCO2arr = new ArrayList();
    static ArrayList snroTmparr = new ArrayList();

    //--VARS USED IN/SINCE THIS PAGE
    string[] xValues = { "CityA", "CityB", "CityC"};


//############################################################################
    static int[] MitArr = new int[3];
    static int[] PopArr = new int[3];

    static double[] DmgPrdT = new double[3];
    static double[] PopPrdT = new double[3];
    static double[] AraPrdT = new double[3];

    static double[] DmgPrdR = new double[3];      // Period Dmg real
    static double[] PopPrdR = new double[3];      // Period Dmg real
    static double[] AraPrdR = new double[3];      // Period Dmg real

    static double[] DmgPrdM = new double[3];      // Period Dmg mitigated
    static double[] PopPrdM = new double[3];      // Period Dmg mitigated
    static double[] AraPrdM = new double[3];      // Period Dmg mitigated
```

```
//##########################################################################
   static double AnnBgt;      // Global variable Annual Budget

   //=======================================================
   // PAGE EVENTS
   //=======================================================
   protected void Page_Init(object sender, EventArgs e)
   {
      Panel1.Visible = Convert.ToBoolean(Session["showMonitor"]);
      //Panel1.Visible = Convert.ToBoolean(1);

      //----------------------------
      // Default btnNext
      //-----------------------------
      btnNext.Enabled = false;
   }

   protected void Page_Load(object sender, EventArgs e)
   {
      if (!IsPostBack)
      {
         //----------------------------------------------------
         // ## Initialize Resource/ Annual Budget
         //----------------------------------------------------
         AnnBgt = getBudget(); Session["AnnBgt"] = AnnBgt;  //# Get annual budget
and pass by session
         lblAnnBgt.Text = Convert.ToString(AnnBgt);
         lblResA.Text = Convert.ToString(0);
         lblResB.Text = Convert.ToString(0);
         lblResC.Text = Convert.ToString(0);
         DropDownList13.SelectedValue = Convert.ToString(0);  //# Default
dropdownlist value

         //------------------------------------------------------
         //%% DEFAULT CHART PROPERTIES
         //------------------------------------------------------
         Chart7.ChartAreas[0].AxisX.MajorGrid.Enabled = false;
         Chart7.ChartAreas[0].AxisY.MajorGrid.Enabled = false;
         //% Set ylabel
         Chart7.ChartAreas[0].AxisY.Title = "% of total budget";
         Chart7.ChartAreas[0].AxisY.TitleFont = new Font("default", 12);
         //% SET CHART LIMIT
         Chart7.ChartAreas[0].AxisX.Maximum = 4;
         Chart7.ChartAreas[0].AxisX.Minimum = -1;
         Chart7.ChartAreas[0].AxisY.Maximum = 100;
         Chart7.ChartAreas[0].AxisY.Minimum = 0;
         //% CREATE DATA POINTS (OTHERWISE, SET UP IN PROPERTY WINDOW)
         Chart7.Series[0].Points.Add(new DataPoint(0, 0) { AxisLabel = "CityA" });
//# THE THIRD POINT HAS TO BE ADDED
```

```
            Chart7.Series[0].Points.Add(new DataPoint(1, 0) { AxisLabel = "CityB" });
//# THE THIRD POINT HAS TO BE ADDED
            Chart7.Series[0].Points.Add(new DataPoint(2, 0) { AxisLabel = "CityC" });
//# THE THIRD POINT HAS TO BE ADDED
            Chart7.Series[0].Points.Add(new DataPoint(3, AnnBgt) { AxisLabel =
"Quota" });   //# THE THIRD POINT HAS TO BE ADDED


            //--INITIALIZE Mit Arr PopArr
            Array.Clear(MitArr, 0, MitArr.Length);
            Array.Clear(PopArr, 0, PopArr.Length);

            //--DISPLAY simYear and nowYear
            simYear=(int)Session["simYear"];
            snroCO2arr = (ArrayList)Session["snroCO2arr"];  //★★★Page05
            snroTmparr = (ArrayList)Session["snroTmparr"];  //★★★Page05

            lblSimYear.Text = Session["simYear"].ToString();    //★★★Page06
            lblNowYear.Text = Session["nowYear"].ToString();    //★★★Default

            //--Display simCO2 SimCO2
            //@@ lblSimCO2.Text = snroCO2arr[simYear - 2000].ToString();
            int curYr = (int)Session["nowYear"];
            double dltCO2 = Convert.ToDouble(snroCO2arr[curYr - 2000]) -
(double)Session["CurCO2"];
            int corCO2 = Convert.ToInt32(Convert.ToDouble(snroCO2arr[simYear - 2000])
- dltCO2);
            lblSimCO2.Text = corCO2.ToString();

            //--Display simTmp SimTmp
            //@@ lblSimTmp.Text = snroTmparr[simYear - 2000].ToString();
            double cordltTmp = Convert.ToDouble(snroTmparr[simYear - 2000]) -
Convert.ToDouble(snroTmparr[curYr - 2000]);
            lblSimTmp.Text = cordltTmp.ToString();

            // PASSING PERIOD STATS
            DmgPrdT = (double[])Session["DmgPrdT"];
            PopPrdT = (double[])Session["PopPrdT"];
            AraPrdT = (double[])Session["AraPrdT"];

            DmgPrdR = (double[])Session["DmgPrdR"];
            PopPrdR = (double[])Session["PopPrdR"];
            AraPrdR = (double[])Session["AraPrdR"];

            //---------------------------------------------
            // Setting Chart Properties (Period Results)
            //---------------------------------------------
            // SHOWING Period damage
            Chart1.Series[0].Points.DataBindY(DmgPrdT);
            Chart2.Series[0].Points.DataBindY(PopPrdT);
            Chart3.Series[0].Points.DataBindY(AraPrdT);
```

```
            Chart1.Series[1].Points.DataBindY(DmgPrdR);
            Chart2.Series[1].Points.DataBindY(PopPrdR);

            //-- Set X-axis Labels
            Chart[] DmgChts = new Chart[3];
            DmgChts[0] = Chart1;
            DmgChts[1] = Chart2;
            DmgChts[2] = Chart3;
            foreach (Chart item in DmgChts)
            {
                item.Series[0].Points[0].AxisLabel = "CityA";
                item.Series[0].Points[1].AxisLabel = "CityB";
                item.Series[0].Points[2].AxisLabel = "CityC";
                //-- Set Gridline
                item.ChartAreas[0].AxisX.MajorGrid.Enabled = false;
                item.ChartAreas[0].AxisY.MajorGrid.Enabled = false;
            }
            //-- Set Legends
            Chart1.Series[0].LegendText = "Occured";
            Chart1.Series[1].LegendText = "Recovered";    //<====Subjected to
change
            Chart1.Legends[0].Docking = Docking.Bottom;
            Chart1.Legends[0].Alignment = StringAlignment.Center;

            Chart2.Series[0].LegendText = "Affected";
            Chart2.Series[1].LegendText = "Helped";       //<====Subjected to
change
            Chart2.Legends[0].Docking = Docking.Bottom;
            Chart2.Legends[0].Alignment = StringAlignment.Center;
            //-- Show Value as label
            Chart1.Series[0].IsValueShownAsLabel = true;
            Chart1.Series[1].IsValueShownAsLabel = true;
            Chart1.Series[1].LabelForeColor = Color.Red;

            Chart2.Series[0].IsValueShownAsLabel = true;
            Chart2.Series[1].IsValueShownAsLabel = true;
            Chart2.Series[1].LabelForeColor = Color.Red;

            Chart3.Series[0].IsValueShownAsLabel = true;
        }
    }

    //=========================================================
    // WEB CONTROL EVENTS
    //=========================================================
    protected void DropDownList13_SelectedIndexChanged(object sender, EventArgs e)
    {
        //--------------------------
        // 1. UPDATE MitArr
        //--------------------------
```

```
        //int[] MitArr = new int[3];
        MitArr[0] = Convert.ToInt32(Convert.ToDouble(DropDownList13.SelectedValue)
* 0.01 * AnnBgt);
        MitArr[1] = Convert.ToInt32(Convert.ToDouble(DropDownList14.SelectedValue)
* 0.01 * AnnBgt);
        MitArr[2] = Convert.ToInt32(Convert.ToDouble(DropDownList15.SelectedValue)
* 0.01 * AnnBgt);
        //■■■
        //lblAnnBgtChk.Text = Convert.ToString(AnnBgt);
        lblResA.Text = Convert.ToString(MitArr[0]);
        lblResB.Text = Convert.ToString(MitArr[1]);
        lblResC.Text = Convert.ToString(MitArr[2]);

        //---------------------------------------
        // 2. DATA VALIDATION (!warning message)
        //---------------------------------------
        int TotMit;
        TotMit = MitArr.Sum();

        if (TotMit > AnnBgt)
        {
            lblDspResAll.Attributes.Add("style", "color:Red");
            lblDspResAll.Text = "You don't have enough budget to use.";
            btnNext.Enabled = false;
        }
        else if (TotMit == AnnBgt)
        {
            lblDspResAll.Attributes.Add("style", "color:blue");
            lblDspResAll.Text = "You've used ALL the budget.";
            btnNext.Enabled = true;
        }
        else
        {
            lblDspResAll.Attributes.Add("style", "");
            lblDspResAll.Text = "You still have extra budget " +
Convert.ToString(AnnBgt - TotMit) + " million $ to use.";
            btnNext.Enabled = true;
        }
        //----------------------------------------------------
        // 3. UPDATING THE CHART (Showing value in percentage)
        //----------------------------------------------------
        Chart7.Series[0].Points[0].SetValueY(DropDownList13.SelectedValue);
        Chart7.Series[0].Points[1].SetValueY(DropDownList14.SelectedValue);
        Chart7.Series[0].Points[2].SetValueY(DropDownList15.SelectedValue);
        Chart7.Series[0].Points[3].SetValueY(100 -

Convert.ToDouble(DropDownList13.SelectedValue) -

Convert.ToDouble(DropDownList14.SelectedValue) -

Convert.ToDouble(DropDownList15.SelectedValue));
```

```
        //--------------------------
        // Store var in Session
        //--------------------------
        Session["MitArr"] = MitArr;        //★★★ Store Mitigation array as
dollars
    }

    //=======================================================
    // User Defined Method
    //=======================================================
    private double getBudget()
    {
        int seed = (int)DateTime.Now.Ticks;
        Random r = new Random(seed);
        double annbgt = r.Next(8, 11) * 10;
        return annbgt;
    }

    protected void Page_Unload(object sender, EventArgs e)
    {
        //--SAVE VARS IN SESSIONS(UPDATE)
        //Session["MitArr"] = MitArr;
        //Session["PopArr"] = PopArr;
    }

    //=======================================================
    // TO NEXT PAGE
    //=======================================================
    protected void btnNext_Click(object sender, EventArgs e)
    {
        //--UPDATE Session["simYear"];
        Session["simYear"] = Convert.ToInt32(Session["simYear"]) + 1;

        //----------------
        // To next page
        //----------------
        Response.Redirect("~/Page06.aspx");
    }
}
```

H.9. Final.aspx

```
//=============================================================================
// Purpose:
//     Showing the statistics of damage , affected population, flooded area,
//     hurricane central pressure records of all teh simulation years.
//=============================================================================

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
//-- add
using System.Collections;
using System.Drawing;
using System.Web.UI.DataVisualization.Charting;

public partial class Default2 : System.Web.UI.Page
{

//################################################################################
###
    // Create Data
    static List<Int32> YrList=new List<Int32>();


//################################################################################
###
    static string[] CityNames=new string[3];//{"CityA","CityB","CityC"};
    static int NowYear = new int();

//################################################################################
###

    static ArrayList DmgSrsA = new ArrayList(); // Series Dmg
    static ArrayList DmgSrsB = new ArrayList(); // Series Dmg
    static ArrayList DmgSrsC = new ArrayList(); // Series Dmg
    static ArrayList DmgSrsAr = new ArrayList(); // Series Dmg reduced
    static ArrayList DmgSrsBr = new ArrayList(); // Series Dmg reduced
    static ArrayList DmgSrsCr = new ArrayList(); // Series Dmg reduced

    static ArrayList PopSrsA = new ArrayList(); // Series Pop
    static ArrayList PopSrsB = new ArrayList(); // Series Pop
    static ArrayList PopSrsC = new ArrayList(); // Series Pop
    static ArrayList PopSrsAr = new ArrayList(); // Series Pop reduced
    static ArrayList PopSrsBr = new ArrayList(); // Series Pop reduced
    static ArrayList PopSrsCr = new ArrayList(); // Series Pop reduced
```

```
    static ArrayList AraSrsA = new ArrayList(); // Series Srs
    static ArrayList AraSrsB = new ArrayList(); // Series Srs
    static ArrayList AraSrsC = new ArrayList(); // Series Srs

    static ArrayList CpTSrs = new ArrayList(); // Series Srs Theoretical Cp
    static ArrayList CpRSrs = new ArrayList(); // Series Srs True Cp


//###############################################################################
###
    //==============================================
    // PAGE EVENTS
    //==============================================

    protected void Page_Init(object sender, EventArgs e)
    {
        Panel1.Visible = Convert.ToBoolean(Session["showMonitor"]);
    }

    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            //----------------------------------------------------
            // Retrive Data
            //----------------------------------------------------
            CityNames = (String[])Session["CityNames"];     //# Retrieve CityName
            NowYear = (int)Session["nowYear"];            //# Retrieve Current Year

            //----------------------------------------------------
            // Create Data
            //----------------------------------------------------
            YrList.Clear();
            for (int i = NowYear; i < 2100+1; i++)
            {
                YrList.Add(i);
            }
            //■ checking
            lblYr.Text = YrList[0].ToString();

            //--ArrayList
            DmgSrsA = (ArrayList)Session["DmgSrsA"];    //★★★
            DmgSrsB = (ArrayList)Session["DmgSrsB"];    //★★★
            DmgSrsC = (ArrayList)Session["DmgSrsC"];    //★★★
            DmgSrsAr = (ArrayList)Session["DmgSrsAr"];    //★★★
            DmgSrsBr = (ArrayList)Session["DmgSrsBr"];    //★★★
            DmgSrsCr = (ArrayList)Session["DmgSrsCr"];    //★★★
            //int[] DmgSrsArInt = Array.ConvertAll(DmgSrsA.ToArray(), o =>
(int)(double)o);

            PopSrsA = (ArrayList)Session["PopSrsA"];    //★★★
```

```
PopSrsB = (ArrayList)Session["PopSrsB"];    //★★★
PopSrsC = (ArrayList)Session["PopSrsC"];    //★★★
PopSrsAr = (ArrayList)Session["PopSrsAr"];    //★★★
PopSrsBr = (ArrayList)Session["PopSrsBr"];    //★★★
PopSrsCr = (ArrayList)Session["PopSrsCr"];    //★★★

AraSrsA = (ArrayList)Session["AraSrsA"];    //★★★
AraSrsB = (ArrayList)Session["AraSrsB"];    //★★★
AraSrsC = (ArrayList)Session["AraSrsC"];    //★★★

//-----------------------------------------------
// Label Value
//-----------------------------------------------
lblNowYr0.Text = NowYear.ToString();

//-----------------------------------------------
//--Chart 1-3: SHOWING SERIES DAMAGE
//-----------------------------------------------
//--Chart1

//Chart1.Series[0].Points.DataBindY(DmgSrsAr);
Chart1.ChartAreas[0].AxisX.Minimum = NowYear;
Chart1.ChartAreas[0].AxisX.Maximum = 2100;
Chart1.ChartAreas[0].AxisY.Title = "Damage(million $)";
Chart1.ChartAreas[0].AxisY.TitleFont = new Font("default", 11);
Chart1.Series[0].Points.DataBindXY(YrList, DmgSrsA);
Chart1.Series[1].Points.DataBindXY(YrList, DmgSrsB);
Chart1.Series[2].Points.DataBindXY(YrList, DmgSrsC);

// Set chart1 legend text
for (int i = 0; i < CityNames.Length; i++)
{ Chart1.Series[i].LegendText = CityNames[i]; }

//--Chart2
Chart2.ChartAreas[0].AxisX.Minimum = NowYear;
Chart2.ChartAreas[0].AxisX.Maximum = 2100;
Chart2.ChartAreas[0].AxisY.Title = "Affected Population";
Chart2.ChartAreas[0].AxisY.TitleFont = new Font("default", 12);
Chart2.Series[0].Points.DataBindXY(YrList,PopSrsA);
Chart2.Series[1].Points.DataBindXY(YrList, PopSrsB);
Chart2.Series[2].Points.DataBindXY(YrList, PopSrsC);
// Set chart1 legend text
for (int i = 0; i < CityNames.Length; i++)
{ Chart2.Series[i].LegendText = CityNames[i]; }

//--Chart3
Chart3.ChartAreas[0].AxisX.Minimum = NowYear;
Chart3.ChartAreas[0].AxisX.Maximum = 2100;
Chart3.ChartAreas[0].AxisY.Title = "Flood Area(km^2)";
Chart3.ChartAreas[0].AxisY.TitleFont = new Font("default", 12);
Chart3.Series[0].Points.DataBindXY(YrList, AraSrsA);
```

```
Chart3.Series[1].Points.DataBindXY(YrList, AraSrsB);
Chart3.Series[2].Points.DataBindXY(YrList, AraSrsC);
// Set chart1 legend text
for (int i = 0; i < CityNames.Length; i++)
{ Chart3.Series[i].LegendText = CityNames[i]; }

//-----------------------------------------------
//--Chart4: Passing Initial CpT; SHOWING SERIES CENTRAL PRESSURE
//-----------------------------------------------
CpTSrs = (ArrayList)Session["CpTSrs"];      //★★★
CpRSrs = (ArrayList)Session["CpRSrs"];      //★★★

//Chart4.Series[0].Points.DataBindY(CpTSrs);
Chart4.ChartAreas[0].AxisX.Minimum = NowYear;
Chart4.ChartAreas[0].AxisX.Maximum = 2100;
Chart4.ChartAreas[0].AxisX.Title = "Year";
Chart4.ChartAreas[0].AxisX.TitleFont = new Font("default", 12);
Chart4.ChartAreas[0].AxisY.Title = "Central Pressure (mb)";
Chart4.ChartAreas[0].AxisY.TitleFont = new Font("default", 10);
Chart4.Series[1].Points.DataBindXY(YrList,CpRSrs);

// Passing Initial CpT; SHOWING SERIES CENTRAL PRESSURE
//-------------------------------------------------
//--Chart 5-7: Percentage of Damage and Mitigation
//-------------------------------------------------
//----------
//--Chart 5:
//----------
double SumDmgA = new double();
double SumDmgAr = new double();
foreach (double item in DmgSrsA)
{
   SumDmgA += item;
}
foreach (double item in DmgSrsAr)
{
   SumDmgAr += item;
}
string[] SDAx = new string[2] { "Net Damage", "Insured" };
double[] SDAy = new double[2];
SDAy[0] = Math.Floor(SumDmgA - SumDmgAr);
SDAy[1] = Math.Floor(SumDmgAr);

// Charting
Chart5.Series[0].Points.DataBindXY(SDAx, SDAy);
Chart5.Series[0].Label = "#VALY";                   //# Lable Setting
Chart5.Series[0].LabelForeColor = Color.White;
Chart5.Series[0].LegendText = "#VALX (#PERCENT)";
Chart5.Series[0].Font = new Font("default", 13);     // Label font size
Chart5.Titles[0].Text = CityNames[0];                //# Title Setting
Chart5.Titles[0].Font = new Font("default", 13);     // Title font size
```

```
Chart5.Series[0].Palette = ChartColorPalette.Excel;

//----------
//--Chart 6
//----------
double SumDmgB = new double();
double SumDmgBr = new double();
foreach (double item in DmgSrsB)
{
   SumDmgB += item;
}
foreach (double item in DmgSrsBr)
{
   SumDmgBr += item;
}
string[] SDBx = new string[2] { "Net Damage", "Insured" };
double[] SDBy = new double[2];
SDBy[0] = Math.Floor(SumDmgB - SumDmgBr);
SDBy[1] = Math.Floor(SumDmgBr);

// Charting
Chart6.Series[0].Points.DataBindXY(SDBx, SDBy);
Chart6.Series[0].Label = "#VALY";                //# Lable Setting
Chart6.Series[0].LabelForeColor = Color.White;
Chart6.Series[0].Font = new Font("default", 13);    // Label font size
Chart6.Series[0].LegendText = "#VALX (#PERCENT)";
Chart6.Titles[0].Text = CityNames[1];             //# Title Setting
Chart6.Titles[0].Font = new Font("default", 13);    // Title font size
Chart6.Series[0].Palette = ChartColorPalette.Excel;

//----------
//--Chart 7
//----------
double SumDmgC = new double();
double SumDmgCr = new double();
foreach (double item in DmgSrsC)
{
   SumDmgC += item;
}
foreach (double item in DmgSrsCr)
{
   SumDmgCr += item;
}
string[] SDCx = new string[2] { "Net Damage", "Insured" };
double[] SDCy = new double[2];
SDCy[0] = Math.Floor(SumDmgC - SumDmgCr);
SDCy[1] = Math.Floor(SumDmgCr);

// Charting
Chart7.Series[0].Points.DataBindXY(SDCx, SDCy);
Chart7.Series[0].Label = "#VALY";
```

```
Chart7.Series[0].LabelForeColor = Color.White;     //# Lable Setting
Chart7.Series[0].Font = new Font("default", 13);    // Label font size
Chart7.Series[0].LegendText = "#VALX (#PERCENT)";
Chart7.Titles[0].Text = CityNames[2];              //# Title Setting
Chart7.Titles[0].Font = new Font("default", 13);    // Title font size
Chart7.Series[0].Palette = ChartColorPalette.Excel;

//---------------------------------------------------------
//--Chart 8-10: Percentage of Affected Pop and Protected
//---------------------------------------------------------
//----------
//--Chart 8
//----------
double SumPopA = new double();
double SumPopAr = new double();
foreach (double item in PopSrsA)
{
   SumPopA += item;
}
foreach (double item in PopSrsAr)
{
   SumPopAr += item;
}
string[] SPAx = new string[2] { "Net affected population", "Secured" };
double[] SPAy = new double[2];
SPAy[0] = Math.Floor(SumPopA - SumPopAr);
SPAy[1] = Math.Floor(SumPopAr);

// Charting
Chart8.Series[0].Points.DataBindXY(SPAx, SPAy);
Chart8.Series[0].Label = "#VALY";
Chart8.Series[0].LabelForeColor = Color.White;         //# Lable Setting
Chart8.Series[0].Font = new Font("default",13);        // Label font size
Chart8.Series[0].LegendText = "#VALX(#PERCENT{P0})";
Chart8.Titles[0].Text = CityNames[0];                  //# Title Setting
Chart8.Titles[0].Font = new Font("default", 13);       //Title font size
Chart8.Series[0].Palette = ChartColorPalette.SeaGreen;

//----------
//--Chart 9
//----------
double SumPopB = new double();
double SumPopBr = new double();
foreach (double item in PopSrsB)
{
   SumPopB += item;
}
foreach (double item in PopSrsBr)
{
   SumPopBr += item;
}
```

```
string[] SPBx = new string[2] { "Net affected population", "Secured" };
double[] SPBy = new double[2];
SPBy[0] = Math.Floor(SumPopB - SumPopBr);
SPBy[1] = Math.Floor(SumPopBr);

// Charting
Chart9.Series[0].Points.DataBindXY(SPBx, SPBy);
Chart9.Series[0].Label = "#VALY";
Chart9.Series[0].LabelForeColor = Color.White;         //# Lable Setting
Chart9.Series[0].Font = new Font("default", 13);       // Label font size
Chart9.Series[0].LegendText = "#VALX(#PERCENT{P0})";
Chart9.Titles[0].Text = CityNames[1];                  //# Title Setting
Chart9.Titles[0].Font = new Font("default", 13);       //Title font size
Chart9.Series[0].Palette = ChartColorPalette.SeaGreen;

//----------
//--Chart 10
//----------
double SumPopC = new double();
double SumPopCr = new double();
foreach (double item in PopSrsC)
{
   SumPopC += item;
}
foreach (double item in PopSrsCr)
{
   SumPopCr += item;
}
string[] SPCx = new string[2] { "Net affected Population", "Secured" };
double[] SPCy = new double[2];
SPCy[0] = Math.Floor(SumPopC - SumPopCr);
SPCy[1] = Math.Floor(SumPopCr);

// Charting
Chart10.Series[0].Points.DataBindXY(SPCx, SPCy);
Chart10.Series[0].Label = "#VALY";
Chart10.Series[0].LabelForeColor = Color.White;        //# Lable Setting
Chart10.Series[0].Font = new Font("default", 13);      // Label font size
Chart10.Series[0].LegendText = "#VALX(#PERCENT{P0})";
Chart10.Titles[0].Text = CityNames[2];                 //# Title Setting
Chart10.Titles[0].Font = new Font("default", 13);      //Title font size
Chart10.Series[0].Palette = ChartColorPalette.SeaGreen;

//-------------------------------------------------------
//--Chart 11-13: Percentage of 3 Cities on Dmg/Pop/Ara
//-------------------------------------------------------
//----------
//--Chart 11
//----------

string[] DmgChtX = new string[3];
```

```
DmgChtX=CityNames;
double[] DmgChtY = new double[3];
DmgChtY[0] = Math.Floor(SumDmgA);
DmgChtY[1] = Math.Floor(SumDmgB);
DmgChtY[2] = Math.Floor(SumDmgC);

// Charting
Chart11.Series[0].Points.DataBindXY(DmgChtX, DmgChtY);
//Chart11.Series[0].IsValueShownAsLabel = true;
Chart11.Series[0].Label = "#VALY";
Chart11.Series[0].LabelForeColor = Color.Black;
Chart11.Series[0].Font = new Font("default",13);
Chart11.Series[0].LegendText = "#VALX(#PERCENT{P0})";
Chart11.Titles[0].Text = "Damage(Million$)";
Chart11.Titles[0].Font = new Font("default", 13);
Chart11.Series[0].Palette = ChartColorPalette.None;

//----------
//--Chart 12
//----------

string[] PopChtX = new string[3];
PopChtX = CityNames;
double[] PopChtY = new double[3];
PopChtY[0] = Math.Floor(SumPopA);
PopChtY[1] = Math.Floor(SumPopB);
PopChtY[2] = Math.Floor(SumPopC);

// Charting
Chart12.Series[0].Points.DataBindXY(PopChtX, PopChtY);
//Chart11.Series[0].IsValueShownAsLabel = true;
Chart12.Series[0].Label = "#VALY";
Chart12.Series[0].LabelForeColor = Color.Black;
Chart12.Series[0].Font = new Font("default", 13);
Chart12.Series[0].LegendText = "#VALX(#PERCENT{P0})";
Chart12.Titles[0].Text = "Affected Population";
Chart12.Titles[0].Font = new Font("default", 13);
Chart12.Series[0].Palette = ChartColorPalette.None;

//----------
//--Chart 13
//----------

double SumAraA = new double();
double SumAraB = new double();
double SumAraC = new double();
foreach (double item in AraSrsA)
   SumAraA += item;
foreach (double item in AraSrsB)
   SumAraB += item;
foreach (double item in AraSrsC)
```

```
                SumAraC += item;

            string[] AraChtX = new string[3];
            AraChtX = CityNames;
            double[] AraChtY = new double[3];
            AraChtY[0] = Math.Floor(SumAraA);
            AraChtY[1] = Math.Floor(SumAraB);
            AraChtY[2] = Math.Floor(SumAraC);

            // Charting
            Chart13.Series[0].Points.DataBindXY(AraChtX, AraChtY);
            Chart13.Series[0].Label = "#VALY";
            Chart13.Series[0].LabelForeColor = Color.Black;
            Chart13.Series[0].Font = new Font("default", 13);
            Chart13.Series[0].LegendText = "#VALX(#PERCENT{P0})";
            Chart13.Titles[0].Text = "Flood Area(m^2)";
            Chart13.Titles[0].Font = new Font("default", 13);
            Chart13.Series[0].Palette = ChartColorPalette.None;
        }
    }

    protected void Page_Unload(object sender, EventArgs e)
    {

    }

    //================================================
    // GO TO NEXT PAGE
    //================================================
    protected void btnNext_Click(object sender, EventArgs e)
    {
        Response.Redirect("~/Default.aspx");
    }
    protected void Chart11_Load(object sender, EventArgs e)
    {

    }
}
```

APPENDIX I VISHURRICANE USER'S MANUAL

Message to Users

VisHurricane is a web-based game for simulating hurricane events and associated damage. In VisHurricane, there are three cities locating at a coastline, and the hurricane events would cause damage, affected population and flood these cities. The global change that involves the increase of CO^2 concentration and global temperature could change the intensity of hurricanes and cause more devastating damage. The user is allowed to choose the scenarios that define the future CO^2 concentration and temperature level. The simulation is designed to run form current year to year 2100. Before reaching year 2100, the user is allowed to allocate budget to mitigate the impact of hurricane events, e.g. reduce the damage and affected population. The user can set the "period length" of simulation. For example, if you choose 10 years as the period length, the simulation will stop every ten years. Every time when you proceed to next period, you can change the government budget that assigned to the cities. At the end of each period, the user can see the statistics of the damage that hurricane events caused during this period and decide the amount of budget to allocate to each city for next period. After the simulation ends in year 2100, the records of hurricane damage and hurricane characteristics will be displayed. In sum, the chosen climate scenarios could affect how much damage these cities would be posed, and the allocation of budget determines how much damage could be mitigated. The following content will introduce the background of the climate scenarios that is used and guide you through each step of VisHurricane.

1. Future Climate Scenarios

The International governmental Panel on Climate Change (IPCC, 2014a) had developed long-term emission scenarios and kept revising since 1990. These scenarios represent the range of driving force (e.g. demographic development, socio-economic development and technological change) and future greenhouse gas (GHG) emissions, based on extensive assessment of literature, modeling approaches that had been widely discussed and reviewed. According to Knutson and Tuleya (2004) the intensity of hurricane will increase due to sea surface temperature increase. VisHurricane uses the global CO^2 concentration and temperature projection of six scenarios groups in IPCC 3^{rd} and 4^{th} annual reports as the driving force that alters the hurricane intensity. Following are the introduction of the selected scenarios that used in VisHurricane (IPCC, 2014b, p18).

- **A1FI/A1B and A1 scenario:**

The A1 storyline and scenario family describes a future world of very rapid economic growth, global population that peaks in mid-century and declines thereafter, and the rapid introduction of new and more efficient technologies. Major underlying themes are convergence among regions, capacity building and increased cultural and social interactions, with a substantial reduction in regional differences in per capita income. The A1 scenario family develops into three groups that describe alternative directions of technological change in the energy system. The three A1 groups are distinguished by their technological emphasis: fossil-intensive (A1FI), non-fossil energy sources (A1T) or a balance across all sources (A1B) (where balanced is defined as not relying too heavily on one particular energy source, on the assumption that similar improvement rates apply to all energy supply and end use technologies)

- **A2 Scenario:**

The A2 storyline and scenario family describes a very heterogeneous world. The underlying theme is self-reliance and preservation of local identities. Fertility patterns across regions converge very slowly, which results in continuously increasing population. Economic development is primarily regionally oriented and per capita economic growth and technological change more fragmented and slower than other storylines.

- **B1 Scenario:**

The B1 storyline and scenario family describes a convergent world with the same global population, that peaks in mid-century and declines thereafter, as in the A1 storyline, but with rapid change in economic structures toward a service and information economy, with reductions in material intensity and the introduction of clean and resource-efficient technologies. The emphasis is on global solutions to economic, social and environmental sustainability, including improved equity, but without additional climate initiatives.

- **B2 Scenario:**

The B2 storyline and scenario family describes a world in which the emphasis is on local solutions to economic, social and environmental sustainability. It is a world with continuously increasing global population, at a rate lower than A2, intermediate levels of economic development, and less rapid and more diverse technological change than in the B1 and A1 storylines. While the scenario is also oriented towards environmental protection and social equity, it focuses on local and regional levels.

2. Steps of VisHurricane

2.1. Front Page (Default.aspx)

Default page is the portal page, giving the brief introduction of VisHurricane.

Click the “Enter” button to proceed to Page02.aspx.

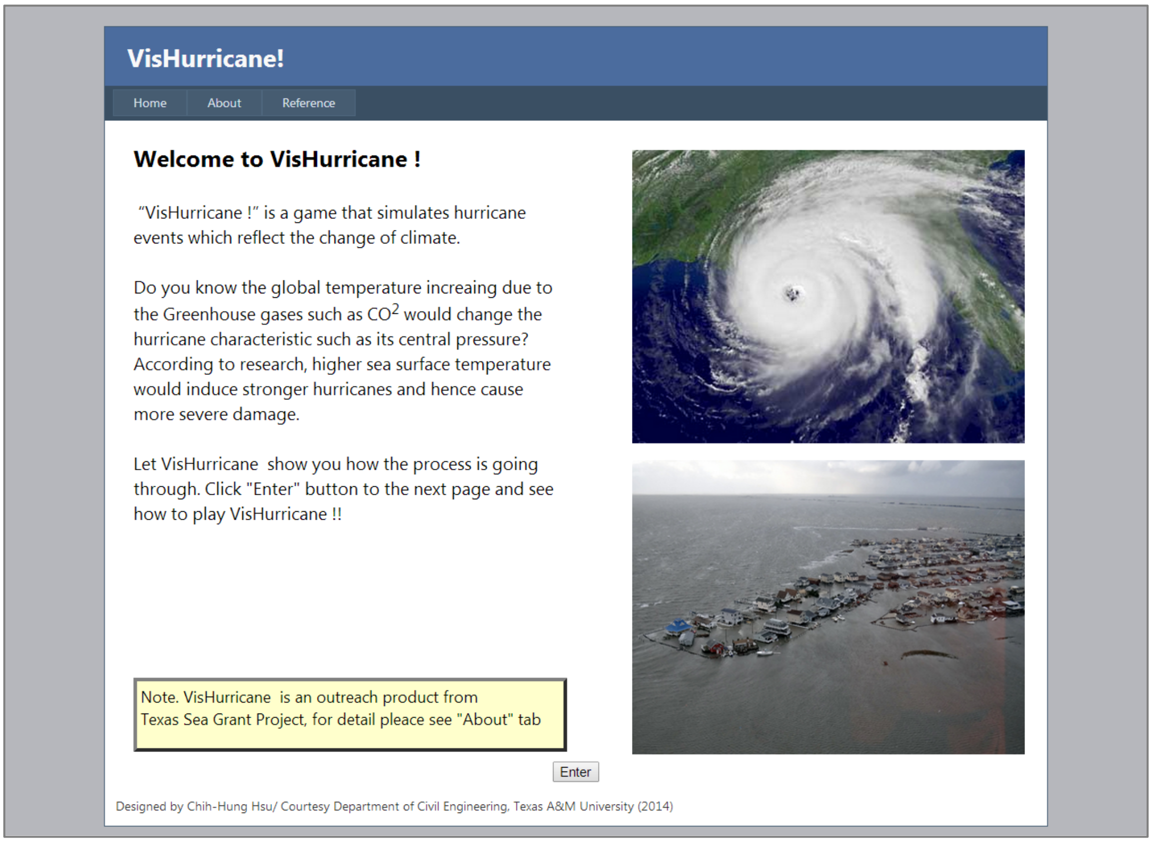


Figure I-1 Default.aspx

Note.

(1) The “About ” tab links to the background of the supporting project.

(2) The “Reference” tab links to the page containing the references about the issues of Climate Change, CO^2 emission and hurricane hazard.

2.2. Page02

Page02.aspx shows the steps of playing VisHurricane.

Click the “Begin” button to proceed to Page03.aspx.

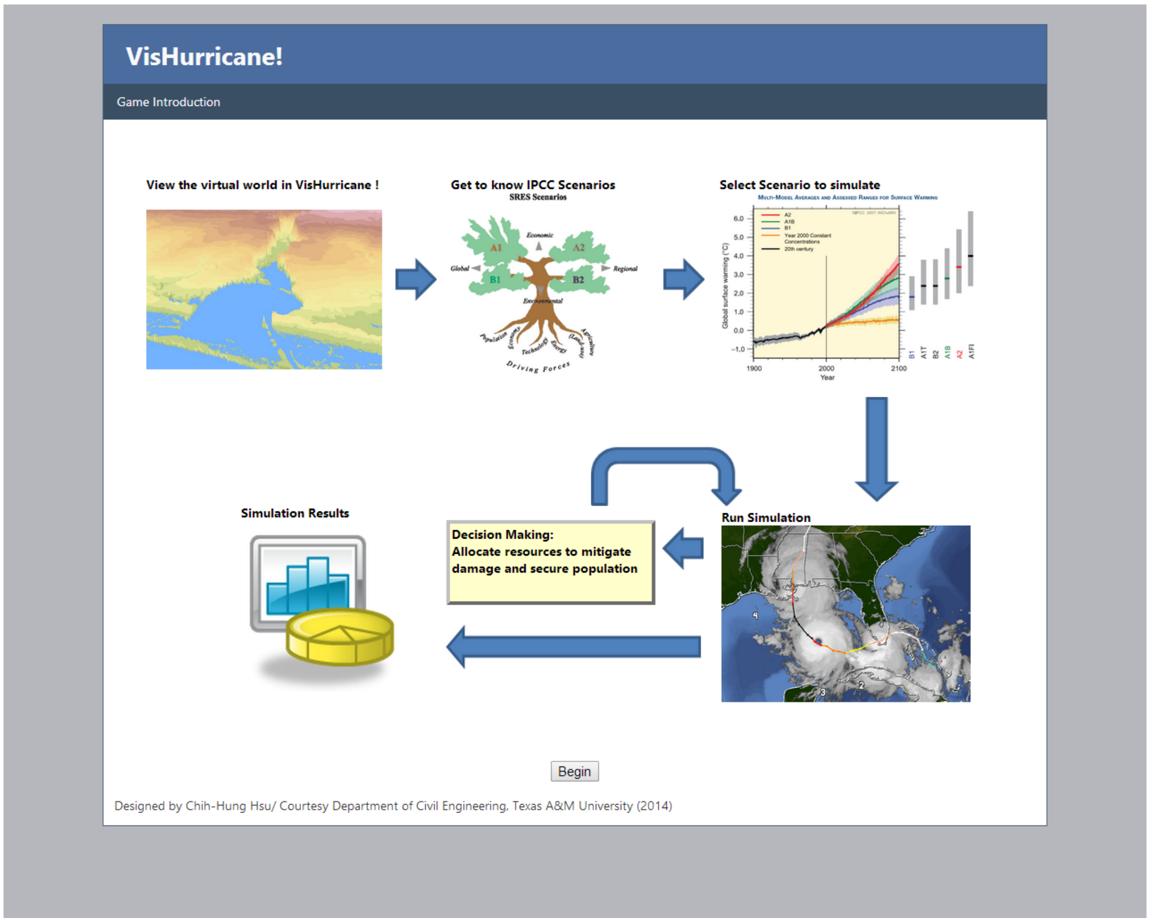


Figure I-2 Page02.aspx

2.3. Page03

Page03.aspx displays the virtual cities and coastline within VisHurricane. You can click the radio button at bottom left to see the image and the description of each city. Click the "Next" button to proceed to Page04.aspx.

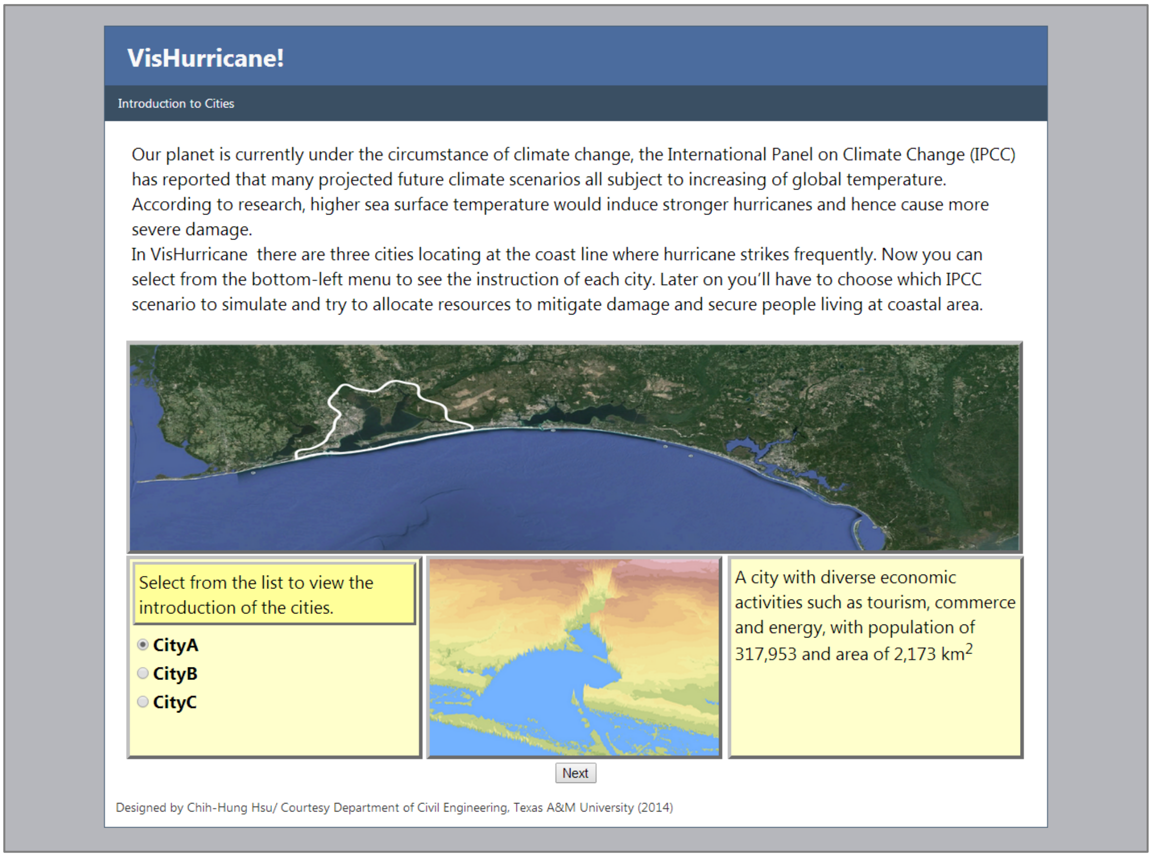


Figure I-3 Page03.aspx

2.4. Page04

Page4.aspx gives the introduction to IPCC future climate scenarios.

Click the radio button at center left to see the introduction of the climate scenarios and the projection for CO2 concentration and temperature.

Click the "Next" button to proceed to Page05.aspx.

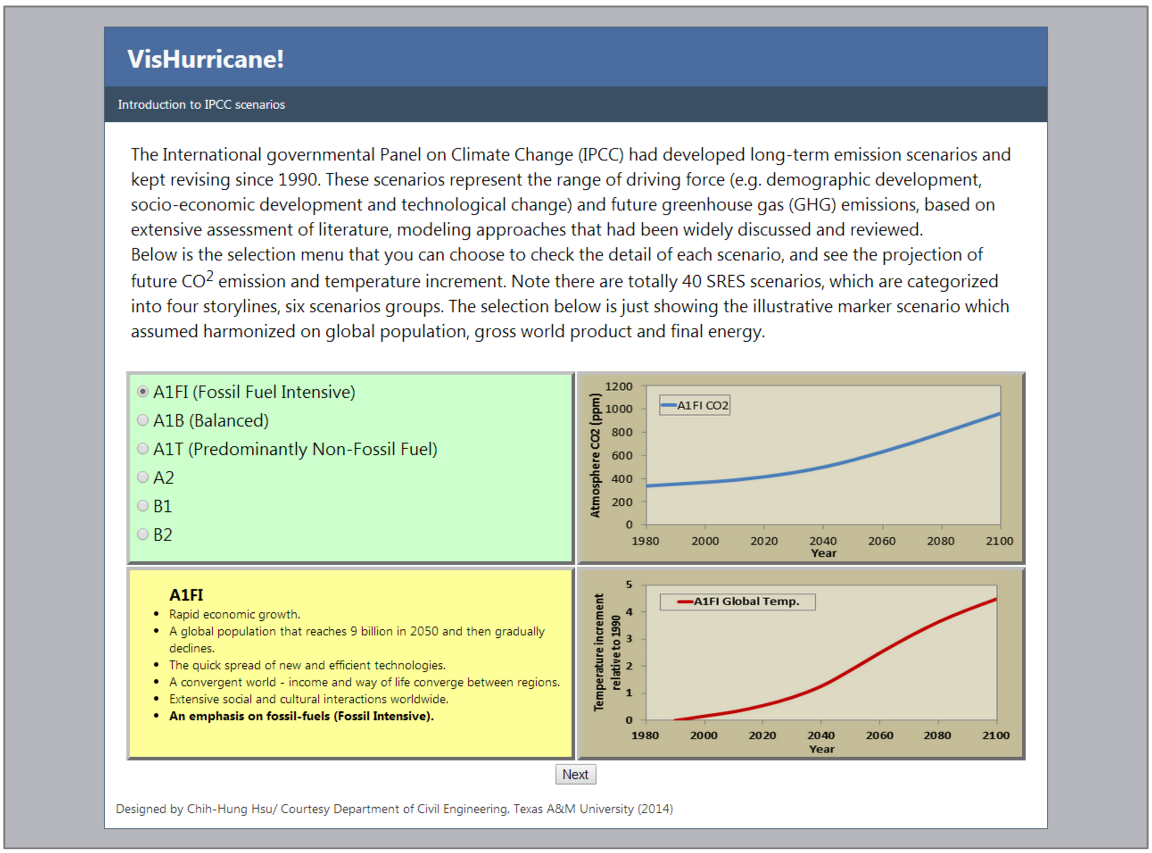


Figure I-4 Page04.aspx

2.5. Page05

In this page you need to make some decisions.

First of all, in the first row of the selection manual, you have to choose the simulation "period length". For example, if you choose 10 years as the "period length", the simulation will run for 10 years without stop. This means you cannot change the setting (the budget you allocate to each city) before the end of this period. Secondly, you can click the radio in the second row of the selection manual to review the projected temperature of the future climate scenarios. Note this will affect the hurricane intensity and result in different levels of damage. Thirdly, you can decide how much budget to allocate to each city. The damage caused by the hurricane events will be reduced according to budget you assigned to each city. Note the budget value shown at the bottom left box is annual budget, which will remain unchanged until the end of each period. Before the simulation runs for next period, the annual budget will be assigned before next period, and you can reallocate the budget to these cities. In the first period you might not know which city is subject to greater damage; however, when this period ends, you'll see charts that show the statistics of the damage caused by hurricane events. You can use the information to help you decide how to allocate the budget for next period of simulation in order to reduce more damage.

Click the "Next" button to proceed to the Mode Selection Page (ModSlt.aspx).

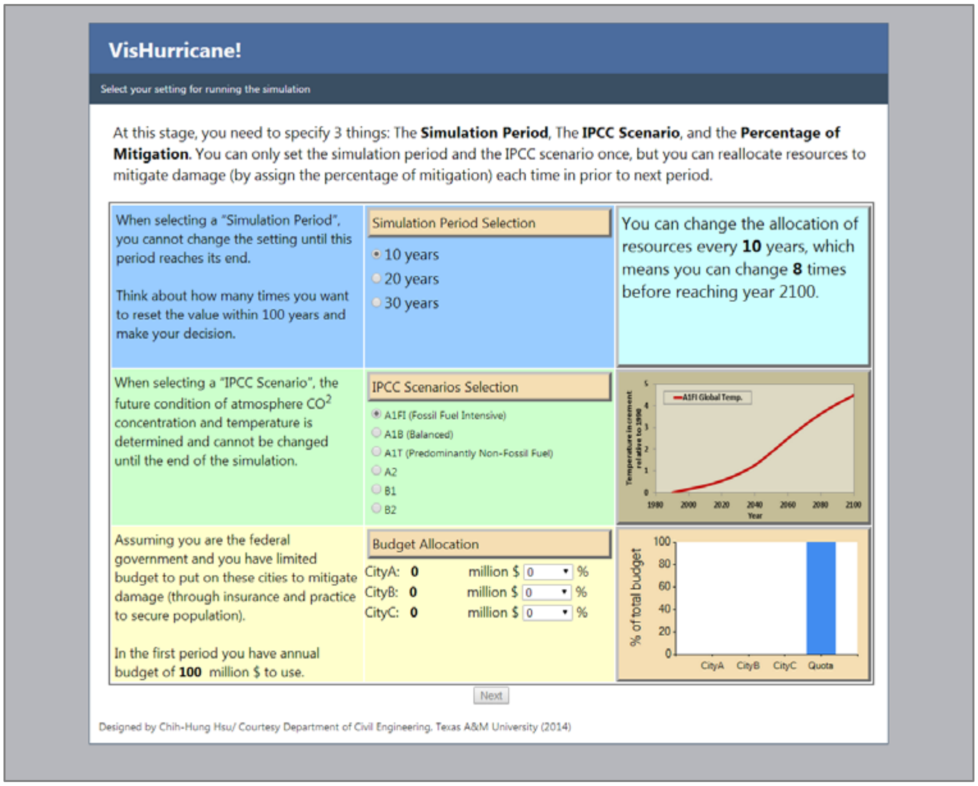


Figure I-5 Page05.aspx

2.6. Mode Selection Page

In this page you can decide how to run the simulation. By clicking “Auto Run”, you allow the computer to run simulation automatically. By clicking “Manual”, you’ll need to click button to proceed the simulation to next year.

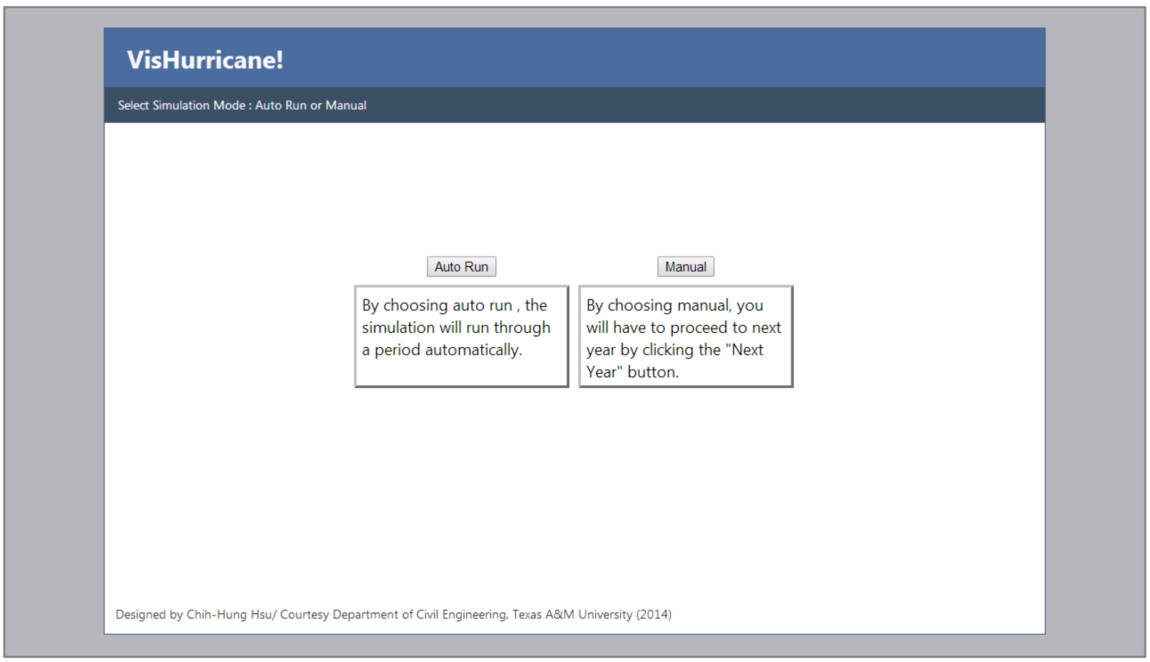


Figure I-6 ModSlt.aspx

2.7. Page06

Page06.aspx is the simulation page. At page top you can see the animation of hurricane striking the coastline if a hurricane event occurs. In the center left box, you will see the information such as current CO2 concentration, global temperature increase. In the bottom left box you will see the hurricane name, category and central pressure for occurred hurricane event. In the bottom right box, you will see the damage caused by the hurricane, population being affected and the flood area.

Click the “Next Year” button to proceed to next year until this period ends.

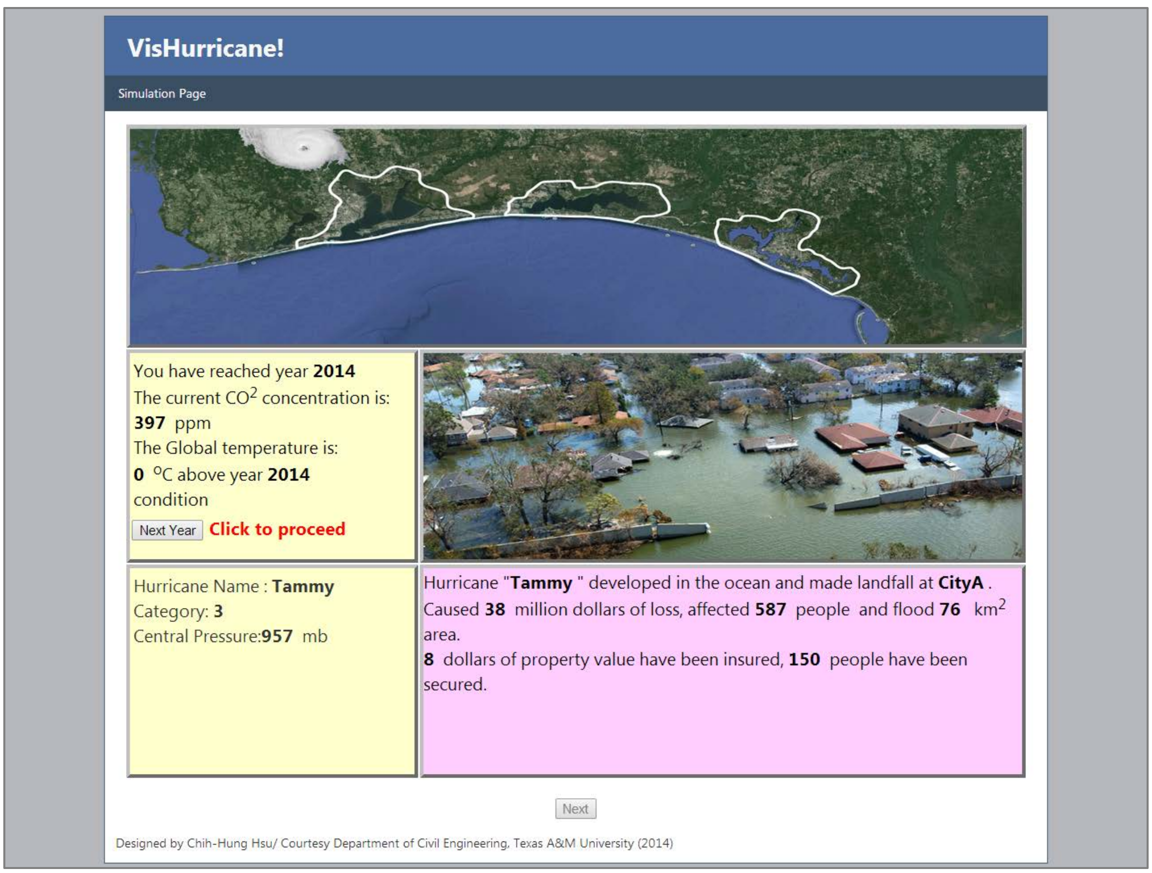


Figure I-7 Page06.aspx

2.8. Page07

Page07.aspx is the setting page before next simulation period. In this page you can see the statistics about pervious period and allowed to reallocate budget before running next period. The damage and affected population is mitigated in proportional to the budget assigned to each city.

Figure I-8 Page07.aspx

2.9. Final Page

The Final.aspx (Figure I-9 through Figure I-11) shows the statistics (damage caused by hurricane, affected population, flooded area, change of hurricane central pressure) of all the simulation years (from starting year to 2100). You might want to try different climate scenarios for hurricane event simulation and try different strategy of allocating the budget to see how the climate factors and the randomness would affect the results.

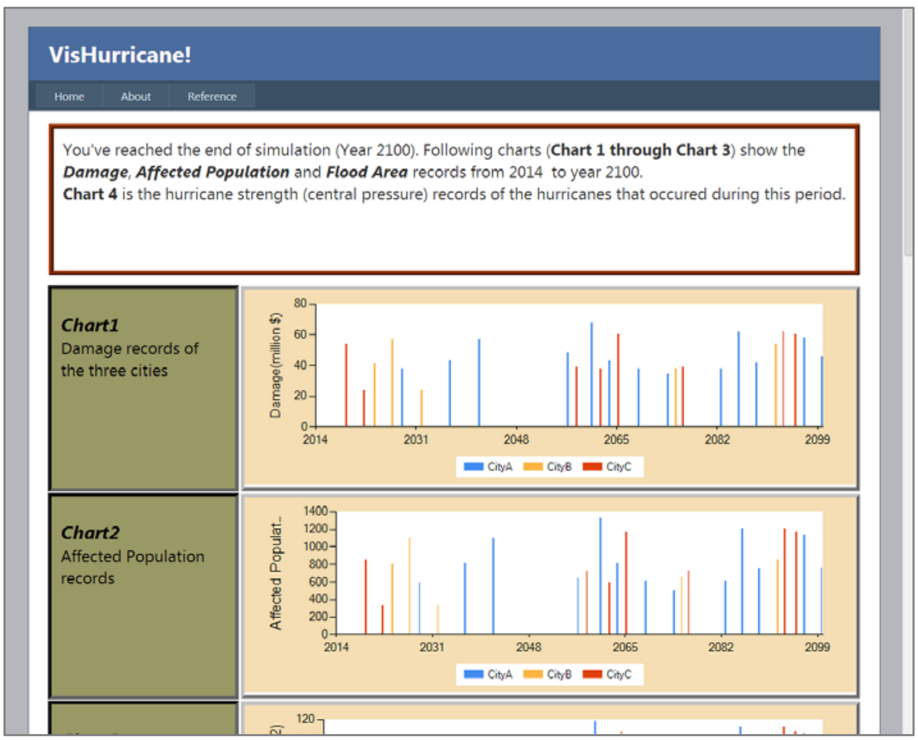


Figure I-9 Final.aspx(1)

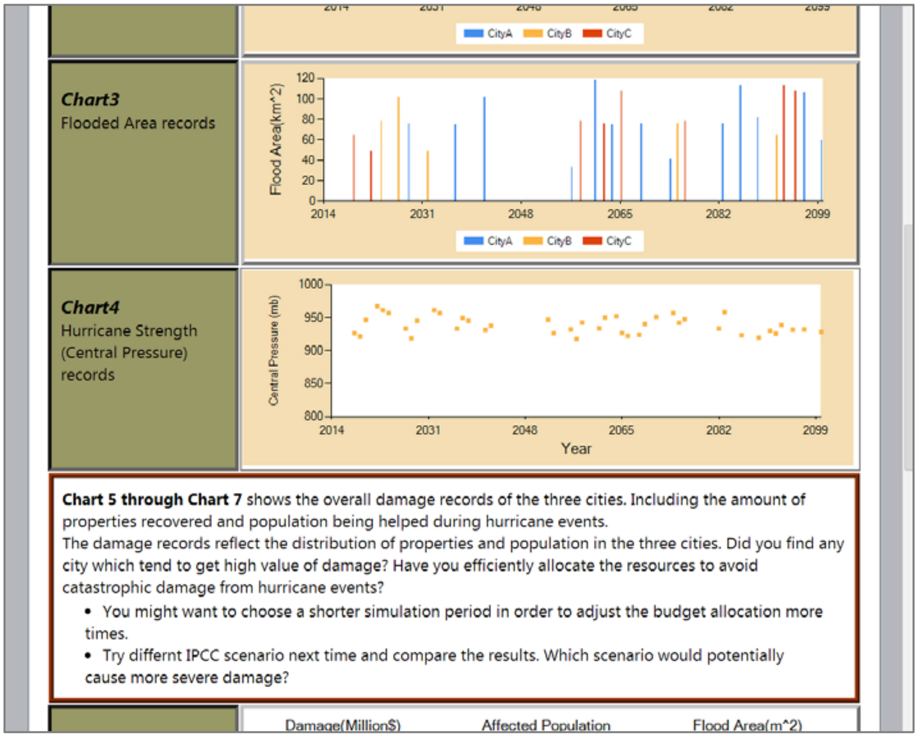


Figure I-10 Final.aspx(2)

Figure I-11 Final.aspx(3)

2.10. About Page

This page shows the information about SeaGrant project and the purpose of creating VisHurricane.

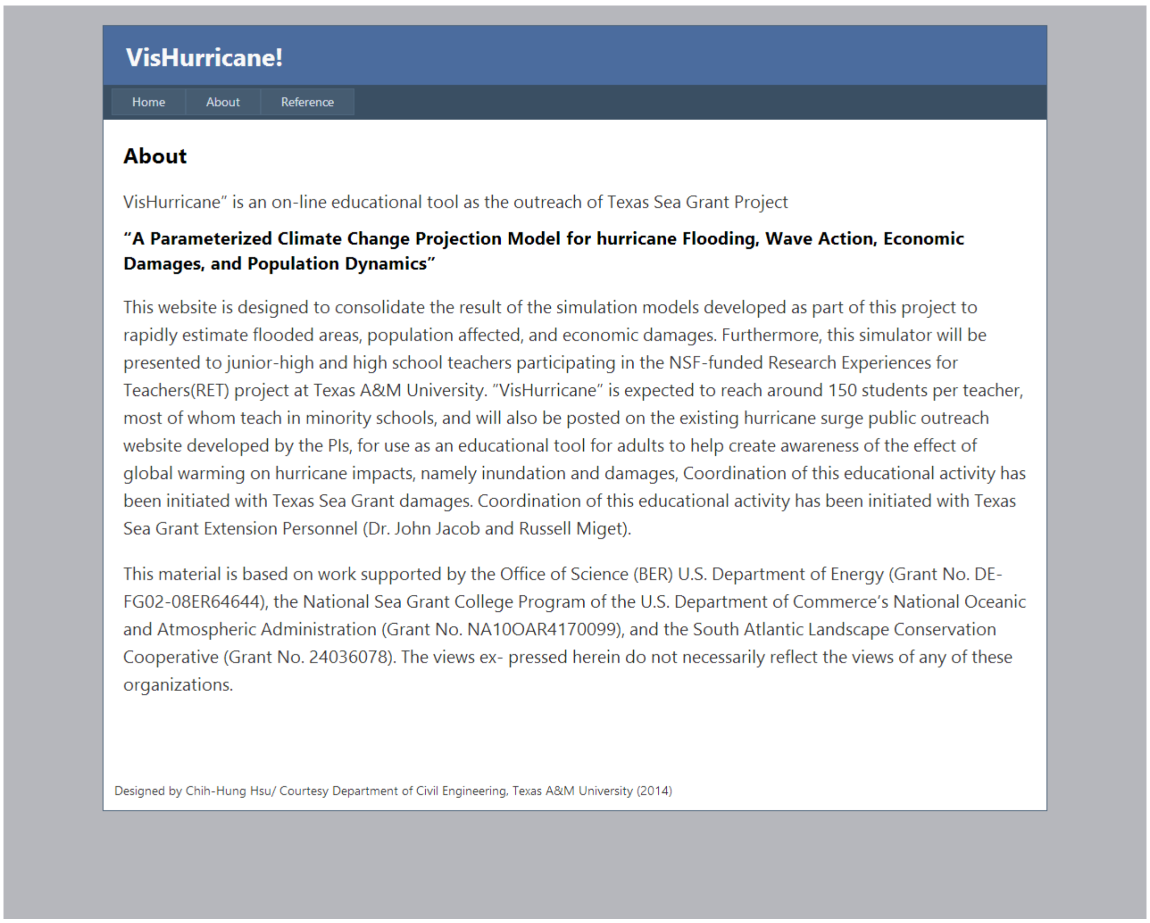


Figure I-12 About.aspx

2.11. Reference Page

Reference.aspx (Figure I-13 and Figure I-14) gives the references about the issues of Climate Change, CO^2 emission and hurricane hazard.

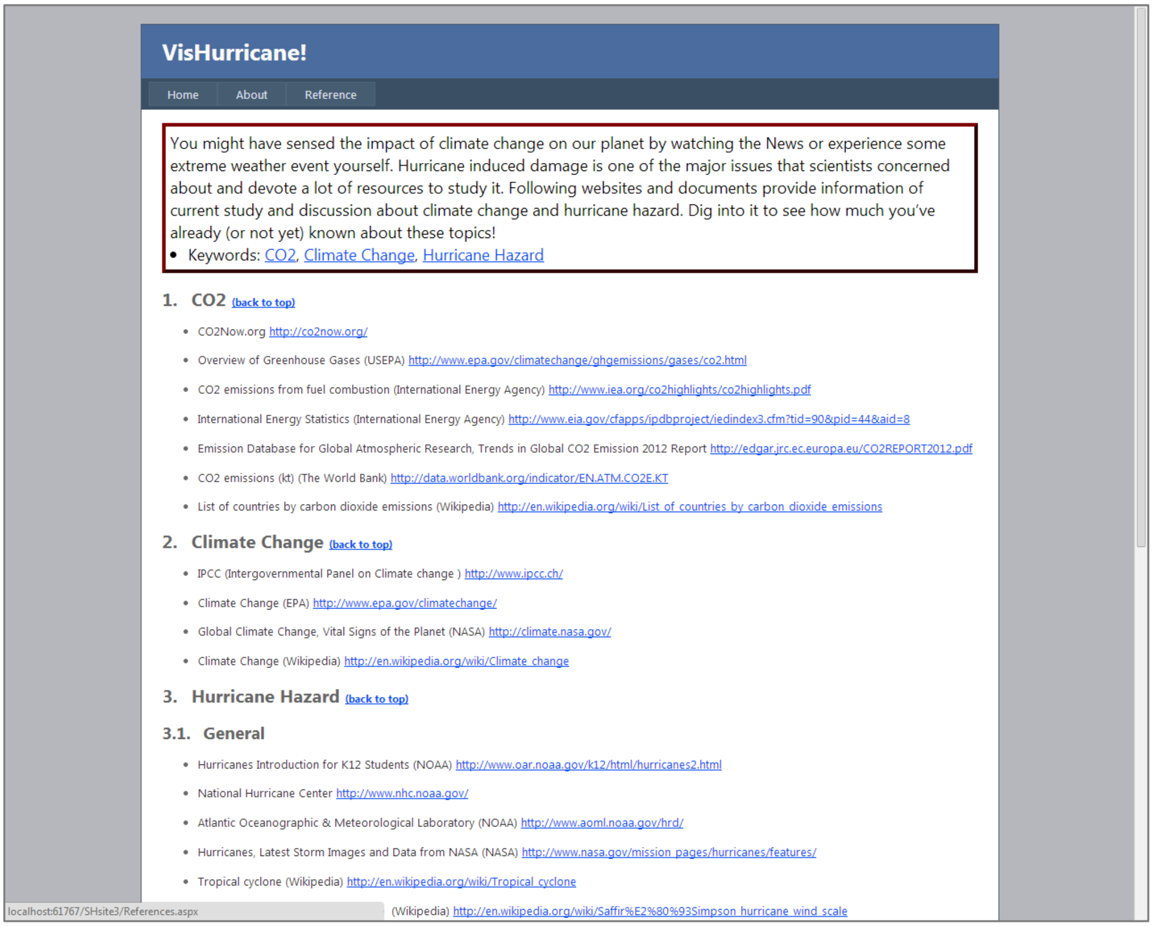
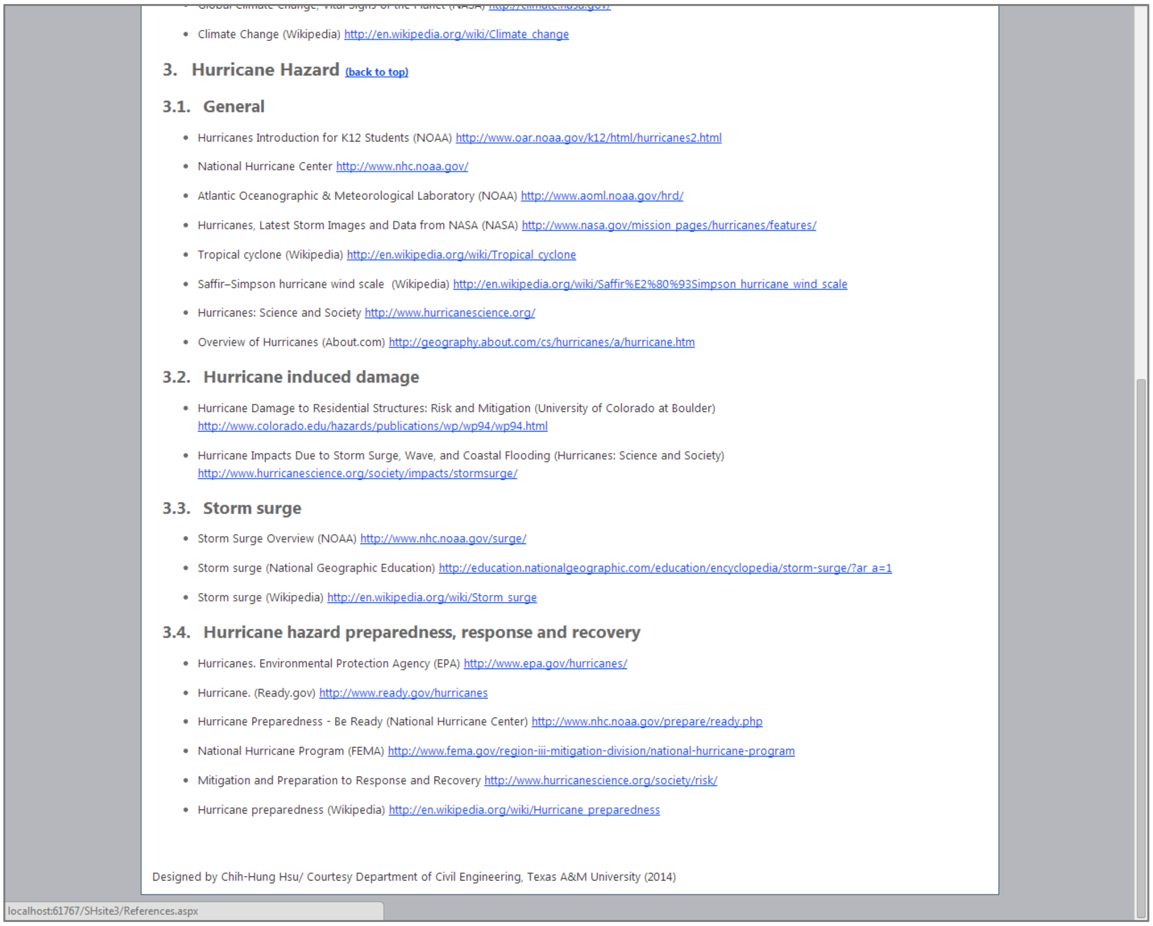


Figure I-13 References.aspx(1)



- Climate Change (Wikipedia) http://en.wikipedia.org/wiki/Climate_change

3. Hurricane Hazard (back to top)

3.1. General

- Hurricanes Introduction for K12 Students (NOAA) http://www.oar.noaa.gov/k12/html/hurricanes2.html
- National Hurricane Center http://www.nhc.noaa.gov/
- Atlantic Oceanographic & Meteorological Laboratory (NOAA) http://www.aoml.noaa.gov/hrd/
- Hurricanes, Latest Storm Images and Data from NASA (NASA) http://www.nasa.gov/mission_pages/hurricanes/features/
- Tropical cyclone (Wikipedia) http://en.wikipedia.org/wiki/Tropical_cyclone
- Saffir–Simpson hurricane wind scale (Wikipedia) http://en.wikipedia.org/wiki/Saffir%E2%80%93Simpson_hurricane_wind_scale
- Hurricanes: Science and Society http://www.hurricanescience.org/
- Overview of Hurricanes (About.com) http://geography.about.com/cs/hurricanes/a/hurricane.htm

3.2. Hurricane induced damage

- Hurricane Damage to Residential Structures: Risk and Mitigation (University of Colorado at Boulder) http://www.colorado.edu/hazards/publications/wp/wp94/wp94.html
- Hurricane Impacts Due to Storm Surge, Wave, and Coastal Flooding (Hurricanes: Science and Society) http://www.hurricanescience.org/society/impacts/stormsurge/

3.3. Storm surge

- Storm Surge Overview (NOAA) http://www.nhc.noaa.gov/surge/
- Storm surge (National Geographic Education) http://education.nationalgeographic.com/education/encyclopedia/storm-surge/?ar_a=1
- Storm surge (Wikipedia) http://en.wikipedia.org/wiki/Storm_surge

3.4. Hurricane hazard preparedness, response and recovery

- Hurricanes. Environmental Protection Agency (EPA) http://www.epa.gov/hurricanes/
- Hurricane. (Ready.gov) http://www.ready.gov/hurricanes
- Hurricane Preparedness - Be Ready (National Hurricane Center) http://www.nhc.noaa.gov/prepare/ready.php
- National Hurricane Program (FEMA) http://www.fema.gov/region-iii-mitigation-division/national-hurricane-program
- Mitigation and Preparation to Response and Recovery http://www.hurricanescience.org/society/risk/
- Hurricane preparedness (Wikipedia) http://en.wikipedia.org/wiki/Hurricane_preparedness

Designed by Chih-Hung Hsu/ Courtesy Department of Civil Engineering, Texas A&M University (2014)

localhost:61767/SHsite3/References.aspx

Figure I-14 References.aspx(2)

2.12. References

IPCC, 2014a. Intergovernmental Panel on Climate Change. https://www.ipcc.ch/index.htm, accessed 2013, April.

IPCC, 2014b. Summary for Policymakers - A report of Working Group I of the Intergovernmental Panel on Climate Change. http://www.ipcc.ch/pdf/assessment-report/ar4/wg1/ar4-wg1-spm.pdf, accessed 2013, June.

Knutson, T. R. and R. E. Tuleya, 2004. Impact of CO2-Induced Warming on Simulated Hurricane Intensity and Precipitation: Sensitivity to the Choice of Climate Model and Convective Parameterization. *Journal of Climate* **17**:3477-3495.

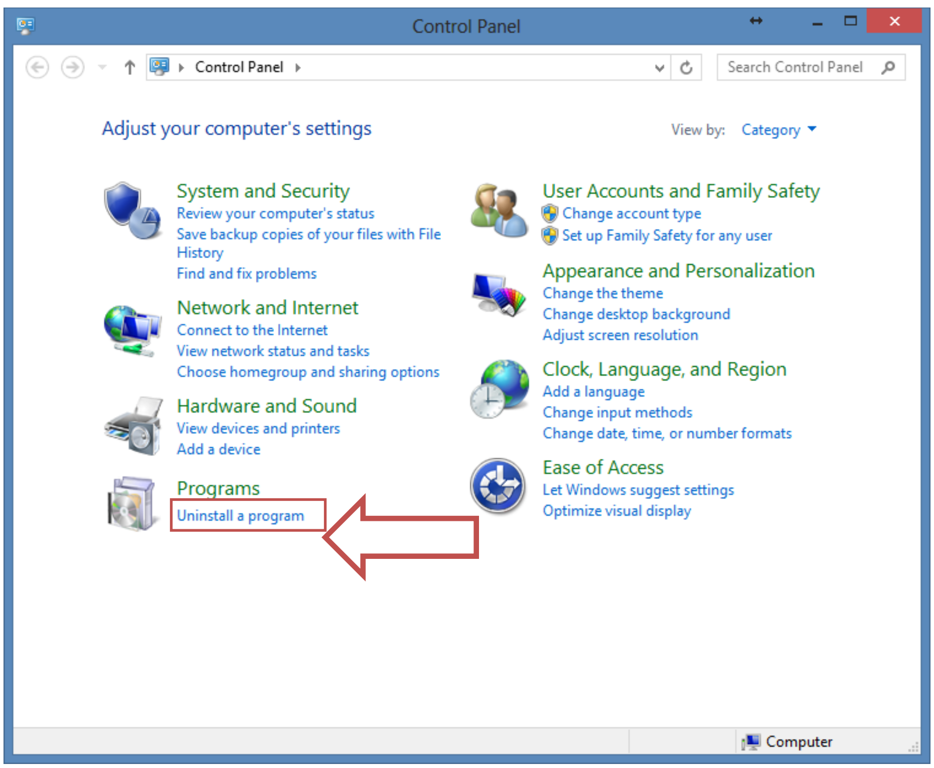
APPENDIX J VISHURRICANE WEBSITE SERVER CONFIGURATION (ON WINDOWS 8)

1. Introduction

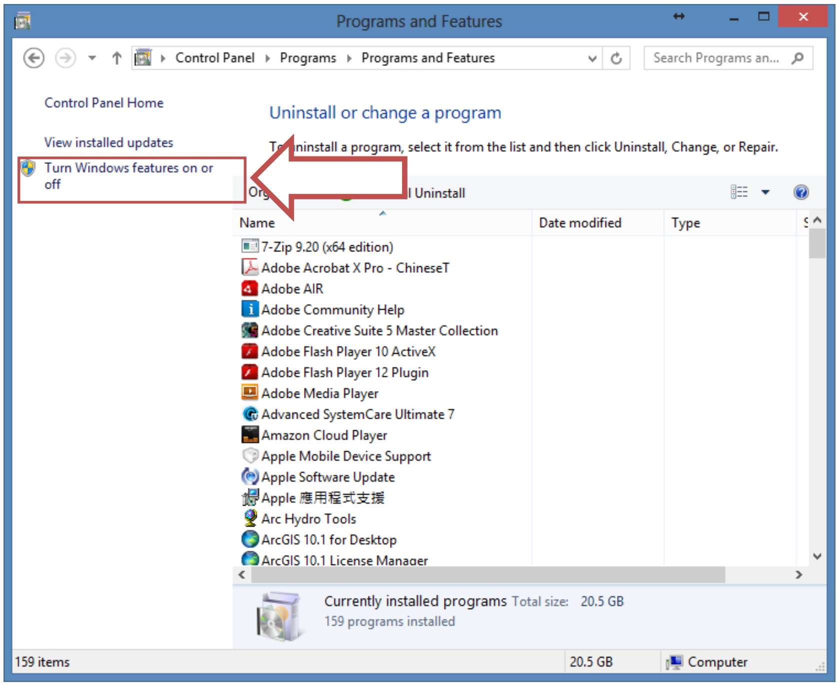
To have the VisHurricane website work on a computer and be accessed on the Internet, a personal computer with a fixed IP address is required. Following are steps to configure the web service under Windows8 operation system and upload VisHurricane files.

2. Steps to configure Internet Information Service (IIS) on Windows 8

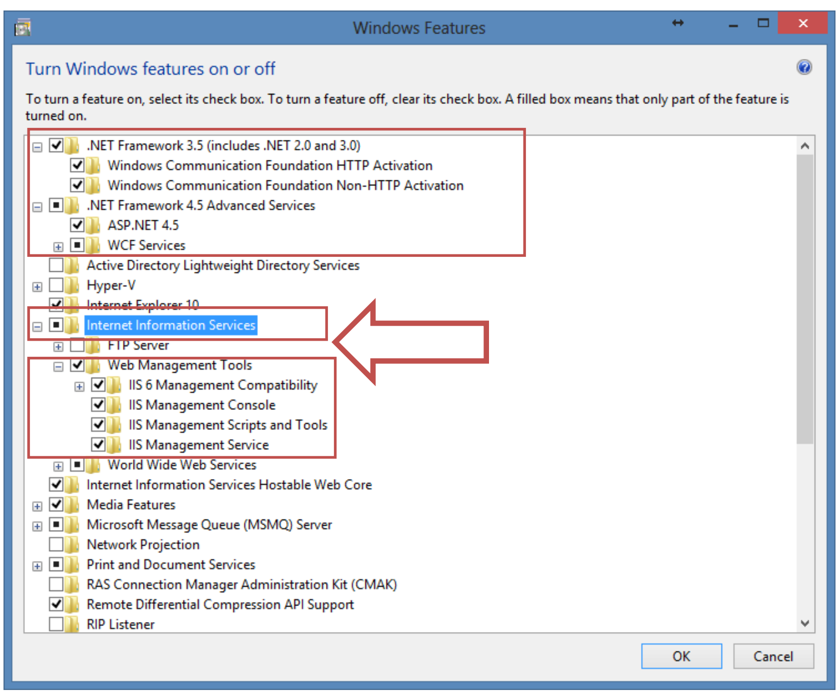
Step1. Turn on Control Panel and Select "Uninstall a Program"



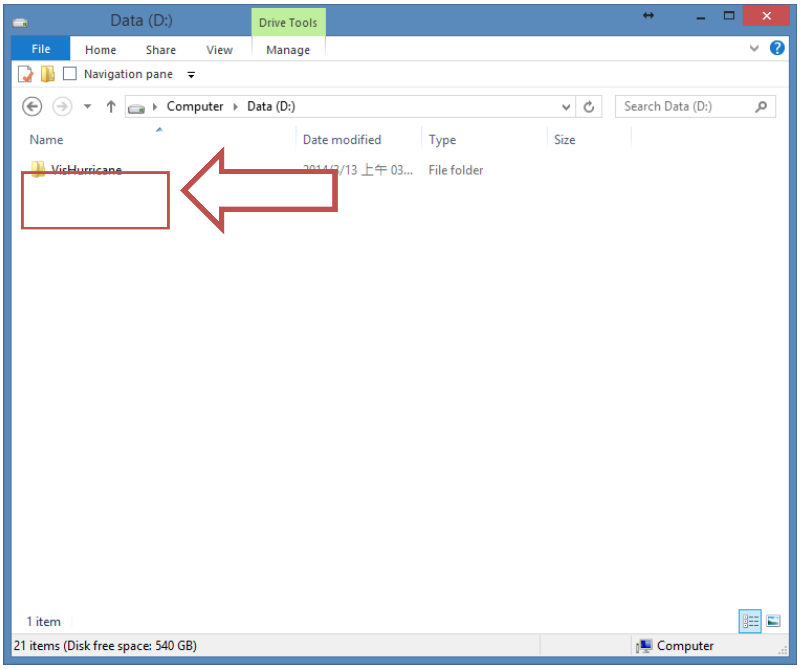
Step2. Select “Turn Windows features on or off”



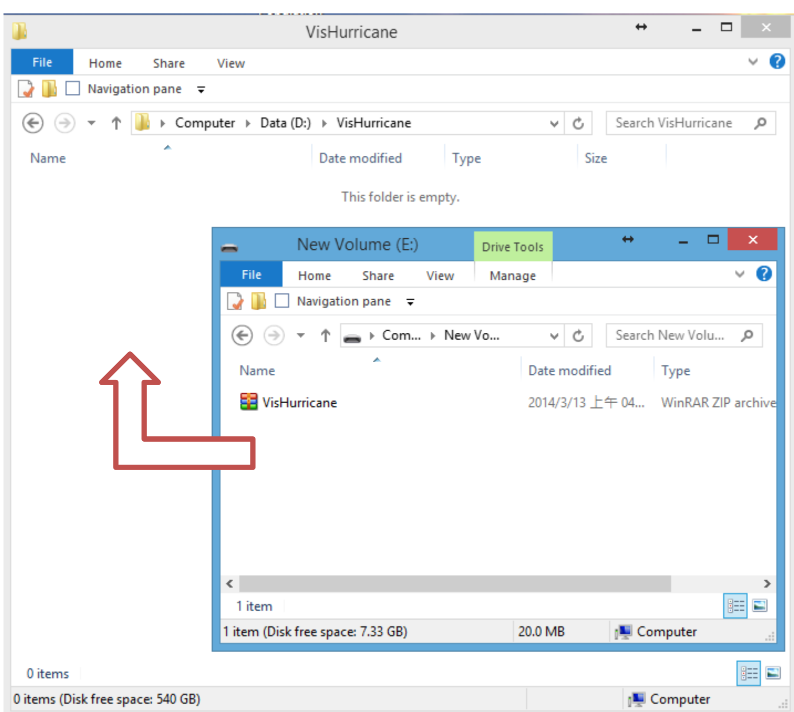
Step3. Turn on “Internet Information Service” and the .NET and IIS 6 features



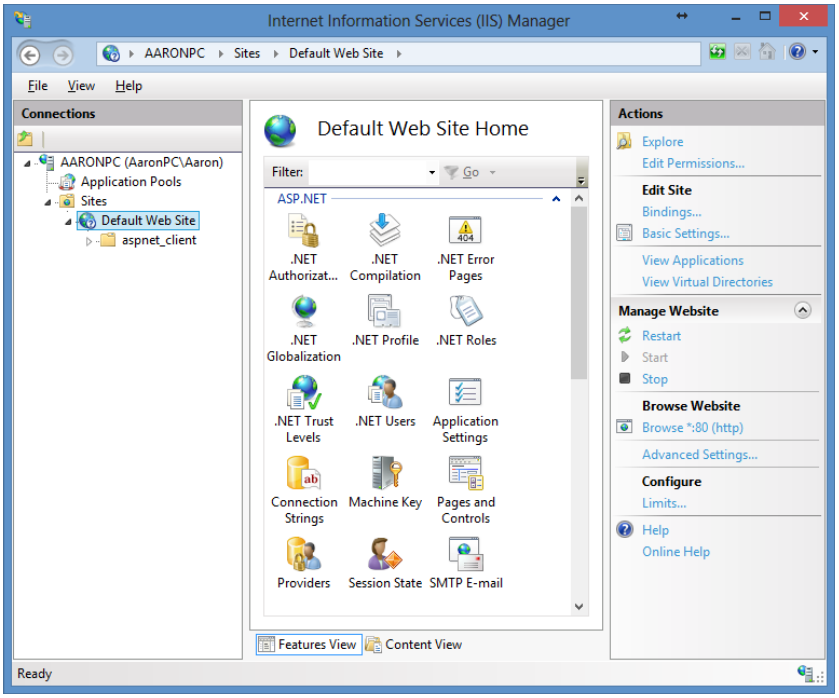
Step4. Create a Folder to Store VisHurricane Files (Here use “VisHurricnae as example”)



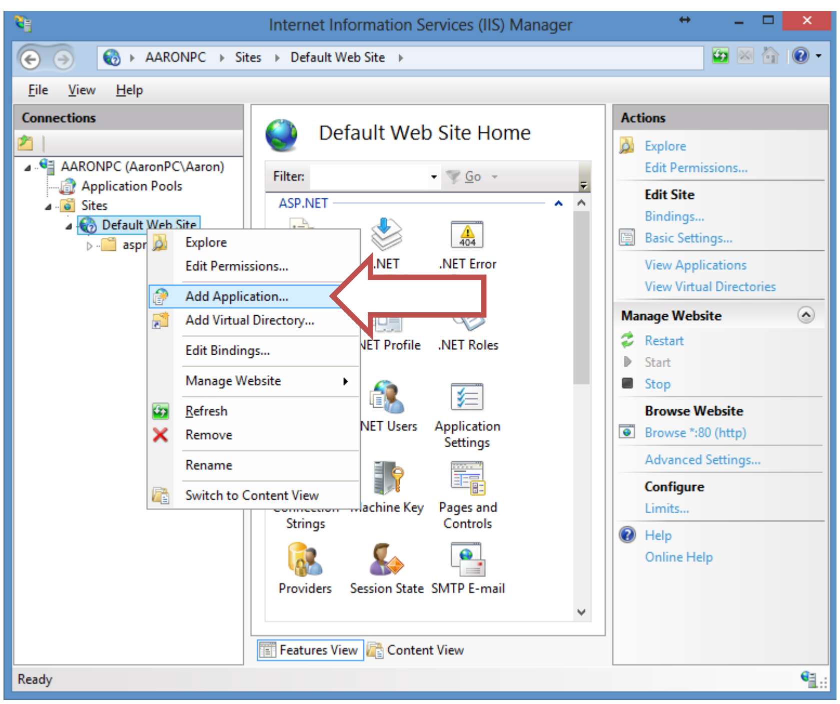
Step5. Unzip the VisHurricane.zip and copy all the files into the folder created in Step4.



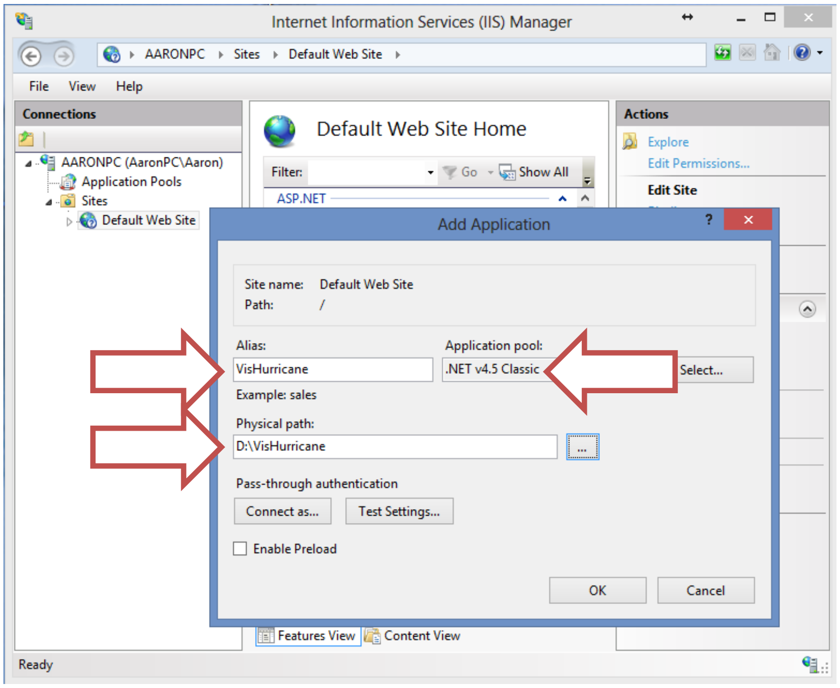
Step6. Turn on Internet Information Services (IIS)



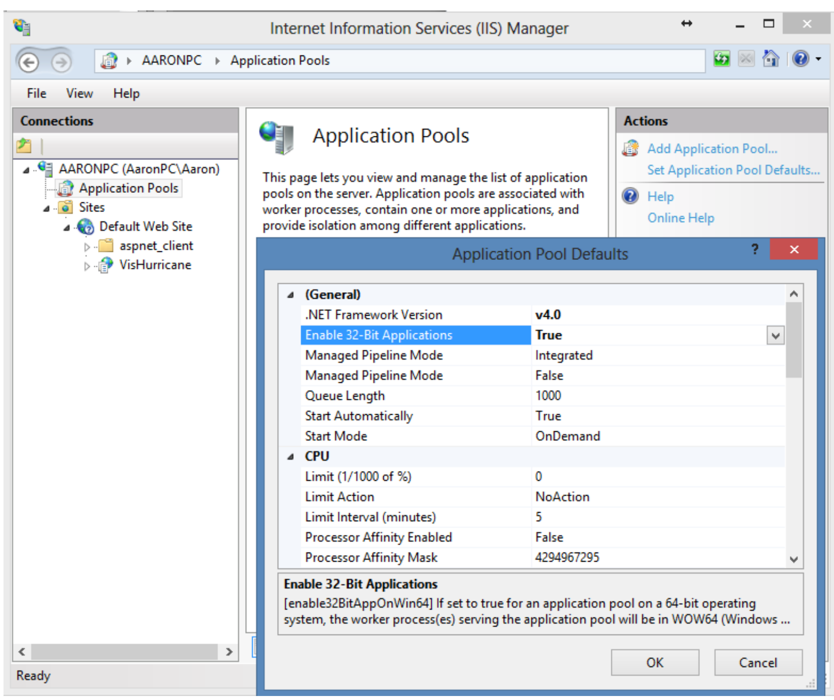
Step7. Right Click on “Default Web Site” and Select “Add Application“



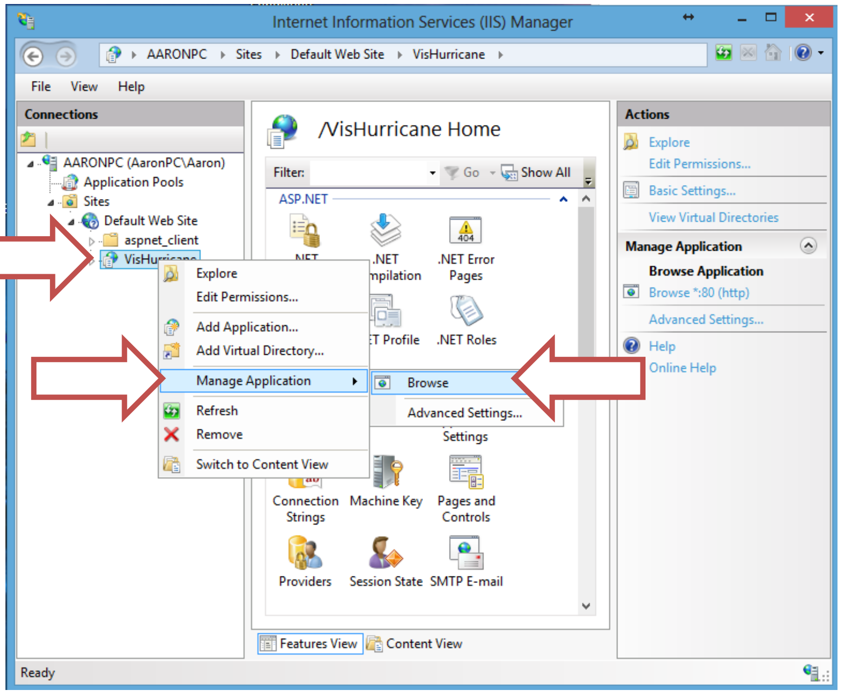
Step8. Give an Alias for the Website and Assign the Physical Path of It (For Application pool, choose .NET v4.5 Classic).



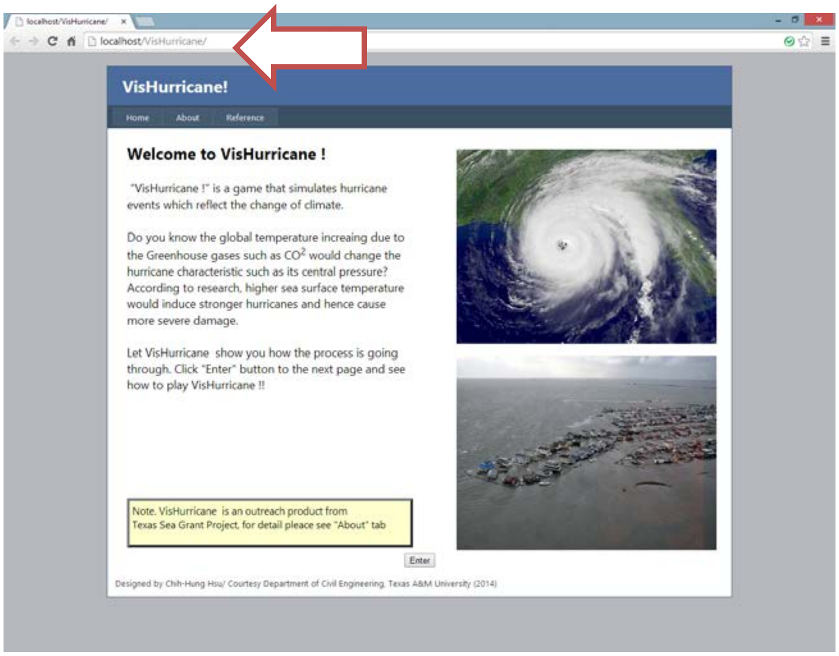
Step9. Click Application Pools→Set Application Pool Defaults...→General, Set "Enable 32-Bit Applications" to "True " mode.



Step10. To Check the website is successfully setup, Right Click “VisHurricane”→Select “Manage Application”→Select “Prowse”



Step11. If “VisHurricane” is successfully setup, the URL address on the browser will show “localhost\VisHurricane”.



Step12. As is mentioned in the beginning, serving the website on the Internet requires a computer with a fixed IP address. Following the steps mentioned above, the URL of VisHurricane will be "(IP address).VisHurricane"